#SHAREorg

SHARE

IBM Ported Tools for z/OS Latest Status and New Features

C.T. Ware
ctware@us.ibm.com
IBM Poughkeepsie, NY

February 4, 2013
Session 12729

SHARE in San Francisco 2013

Trademarks and Disclaimers

- **See http://www.ibm.com/legal/copytrade.shtml for a list of IBM trademarks.**
- **The following are trademarks or registered trademarks of other companies**
  - UNIX is a registered trademark of The Open Group in the United States and other countries
  - CERT® is a registered trademark and service mark of Carnegie Mellon University.
  - ssh® is a registered trademark of SSH Communications Security Corp
  - X Window System is a trademark of X Consortium, Inc
- **All other products may be trademarks or registered trademarks of their respective companies**

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

Agenda

- **New OpenSSH for z/OS Extension**
  - **>> Overview <<**
  - Packaging and installation
  - Usage

- New: sudo for z/OS
  - Overview
  - Packaging and installation
  - Usage
  - Examples
- Appendix

Overview

- **Problem Statement**
  - As an OpenSSH for z/OS user, I want to exploit my z/OS hardware cryptographic support in order to improve the performance of and minimize the CPU time consumed by my SSH sessions on z/OS.
- **Solution**
  - A z/OS extension was added to OpenSSH for z/OS V1R2 for Integrated Cryptographic Service Facility (ICSF) ciphers and MAC (message authentication code) algorithms support.

Overview

- **Benefits**
  - By allowing ICSF to implement certain ciphers and MAC algorithms, OpenSSH for z/OS can use CP Assist for Cryptographic Function (CPACF) hardware cryptographic support when applicable. This support can improve the performance of and minimize the CPU time consumed by SSH sessions on z/OS.
  - Support applies to all of the client and server commands (**ssh**, **scp**, **sftp**, **sshd** and **sftp-server**)

Overview

- **Benefits (Continued)**
  - Internal performance test results, ICSF with CPACF hardware support versus OpenSSL software:
    - ICSF provided a significant reduction in CPU time for the 3des-cbc and aes*-cbc ciphers with the hmac-sha1 MAC algorithm.
    - In general when using ICSF, CPU time reduction increased as the amount of data transferred increased.
    - These are not officially published performance results. Your results may vary.

Overview

- **Miscellaneous requirements that were addressed**
  - UR1 APAR OA36257 – Eliminated the unnecessary SMF error messages.
  - DOC APAR OA34819 – Modified buffer reallocation to minimize heap fragmentation.
  - Added internal serviceability improvements available via the _ZOS_OPENSSH_DEBUG environment variable.

Complete your sessions evaluation online at SHARE.org/SFEval

Agenda

- **New OpenSSH for z/OS Extension**
  - Overview
  - **>> Packaging and installation <<**
  - Usage
- New: sudo for z/OS
  - Overview
  - Packaging and installation
  - Usage
  - Examples
- Appendix

Complete your sessions evaluation online at SHARE.org/SFEval

Packaging and installation

- **This support is provided via the PTF for APAR OA37278 to IBM Ported Tools for z/OS: OpenSSH V1R2 (Product ID 5655-M2301, FMID HOS1120).**
- **z/OS 1.10 and z/OS 1.11 requirement:** ICSF FMID HCR7770 must be installed before enabling the support. ICSF FMID HCR7770 is not a requirement for installing the APAR.
- **Reminder:** OpenSSH for z/OS V1R2 is supported on z/OS 1.10 and later.

SHARE in San Francisco 2013

Packaging and installation

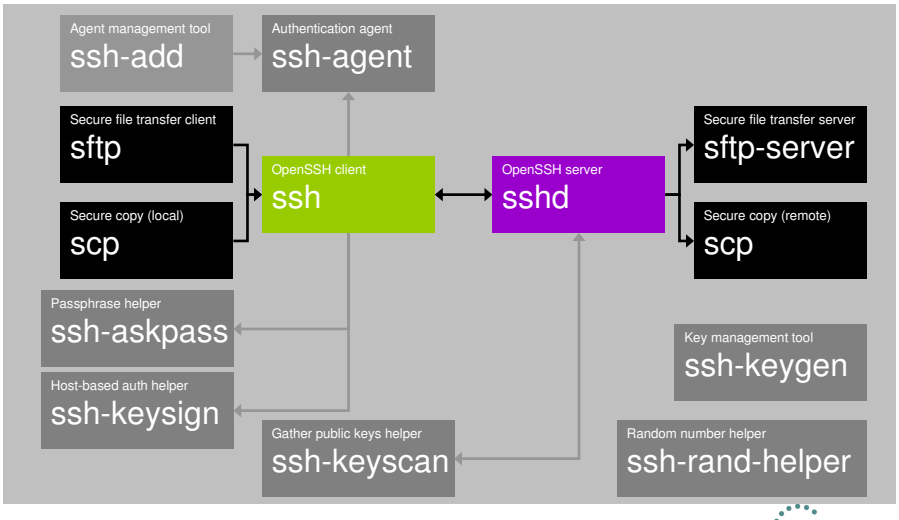
- **Updated OpenSSH for z/OS V1R2 parts:**
  - `/bin/ssh`
  - `/bin/scp`
  - `/bin/sftp`
  - `/bin/ssh-add`
  - `/bin/ssh-agent`
  - `/bin/ssh-keygen`
  - `/bin/ssh-keyscan`
  - `/usr/lib/ssh/ssh-keysign`
  - `/usr/lib/ssh/ssh-rand-helper`
  - `/usr/lib/ssh/sftp-server`
  - `/usr/sbin/sshd`
  - `/usr/lib/nls/msg/C/openssh.cat`
  - `/usr/man/C/man1/fotz200.book`
  - `/samples/ssh_smf.h`
  - `SYS1.MACLIB(FOTSMF77)`

SHARE in San Francisco 2013

Agenda

- **New OpenSSH for z/OS Extension**
  - Overview
  - Packaging and installation
  - **>> Usage <<**
- New: sudo for z/OS
  - Overview
  - Packaging and installation
  - Usage
  - Examples
- Appendix

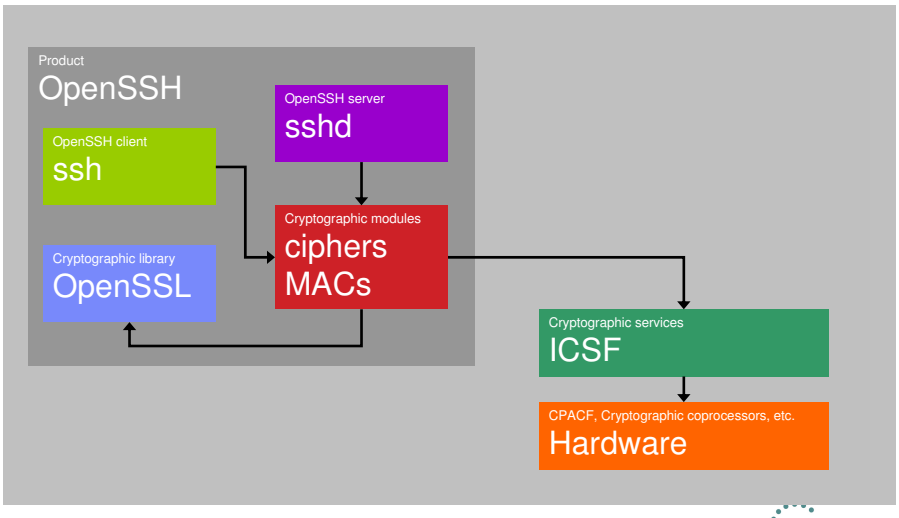
Usage



Usage

- **OpenSSH for z/OS can be set up to use ICSF to implement certain ciphers and MAC algorithms**
  - Enables the use of CPACF hardware support via ICSF.
  - Expands the use of ICSF by OpenSSH for z/OS.
  - Can improve the performance of OpenSSH for z/OS since the ciphers and MAC algorithms represent a significant portion of the processing done during an SSH session.

Usage



Usage

- **Ciphers that can be implemented by ICSF:**
  - aes128-cbc, aes192-cbc, aes256-cbc, 3des-cbc
  - rijndael-cbc@lysator.liu.se (same as aes256-cbc)
  - blowfish-cbc, arcfour, arcfour128, arcfour256
- **Ciphers with CPACF hardware support via ICSF:**
  - aes128-cbc, aes192-cbc, aes256-cbc, 3des-cbc
  - rijndael-cbc@lysator.liu.se (same as aes256-cbc)
- **Ciphers that can NOT be implemented by ICSF (OpenSSL only):**
  - aes128-ctr, aes192-ctr, aes256-ctr
  - cast128-cbc, acss@openssh.org

Usage

- **MAC algorithms that can be implemented by ICSF:**
  - hmac-sha1, hmac-sha1-96
  - hmac-md5, hmac-md5-96
  - hmac-ripemd160, hmac-ripemd160@openssh.com
- **MAC algorithms with CPACF hardware support via ICSF:**
  - hmac-sha1, hmac-sha1-96
- **MAC algorithms that can NOT be implemented by ICSF (OpenSSL only):**
  - umac-64@openssh.com

Usage

- **Steps to set up OpenSSH for z/OS to use ICSF ciphers and MAC algorithms**
  1. Verify that ICSF is started.
  2. Verify that OpenSSH for z/OS users (including the **sshd** privilege separation user and the user that starts the **sshd** daemon) have access to the appropriate profiles in the CSFSERV general resource class.
  3. For the client: Set the new CiphersSource and MACsSource keywords to “any” or “ICSF” in the appropriate z/OS-specific OpenSSH client configuration files, **zos_ssh_config** or **zos_user_ssh_config**.
  4. For the server: Set the new **zos_sshd_config** keywords CiphersSource and MACsSource to “any” or “ICSF”.
  5. Modify the client and server side ciphers and MAC algorithms lists.

More details on the following slides.

SHARE in San Francisco 2013

Usage

- **CSFSERV accesses required for ICSF ciphers (Step 2):**
  - READ access to the CSFIQA, CSF1TRC, CSF1TRD, CSF1SKE and CSF1SKD profiles.
- **CSFSERV accesses required for ICSF MAC algorithms (Step 2):**
  - READ access to the CSFIQA, CSF1TRC, CSF1TRD and CSFOWH profiles.

SHARE in San Francisco 2013

Usage

- **New CiphersSource keyword values (Steps 3 & 4):**
  - "OpenSSL" → Implement ciphers using the statically linked OpenSSL cryptographic library. This is the default.
  - "ICSF" → Implement ciphers using ICSF. Ciphers not supported by ICSF will fail if used.
  - "any" → Implement ciphers using ICSF if available. Ciphers not supported by ICSF will be implemented using OpenSSL. If ICSF isn't available, all ciphers will be implemented using OpenSSL.

Usage

- **New MACsSource keyword values (Steps 3 & 4):**
  - Same as CiphersSource, but applies to MAC algorithms.
- **Locations CiphersSource and MACsSource keywords supported (Steps 3 & 4):**
  - In the **zos_user_ssh_config**, **zos_ssh_config** and **zos_sshd_config** configuration files
  - On the command-line using the **ssh**, **sftp**, **scp** and **sshd** –o options

Usage

- **Modifying the ciphers and MAC algorithms lists (Step 5):**
  - Done via the existing **ssh_config** and **sshd_config** keywords Ciphers and MACs.
  - **Required:** If using “ICSF” source, modify the lists to only contain values supported by ICSF.
  - **Required:** If the CiphersSource keyword is set to "ICSF" and if privilege separation is enabled, remove the arcfour, arcfour128 and arcfour256 ciphers from the server side ciphers list.

Usage

- **(Continued) Modifying the ciphers and MAC algorithms lists (Step 5):**
  - **Required:** If FIPS 140-2 compliance is required and OpenSSH is not exempt from compliance, modify the lists to only contain values supported by ICSF in FIPS 140-2 mode. In addition, the "ICSF" source must be used to ensure ICSF FIPS 140-2 compliant ciphers and MAC algorithms are used.
  - **Optional:** Modify the lists to prefer values with CPACF hardware support via ICSF.

Usage

- **Important notes when modifying the ciphers and MAC algorithms lists (Step 5):**
  - The user's guide provides example lists that adhere to the required and optional modifications.
  - The client selects the cipher and MAC algorithm to use during an SSH session from the lists offered by the server.
  - If the client and server fail to negotiate a cipher or MAC algorithm, the SSH session will end.
  - The client is allowed to choose any cipher and MAC algorithm from the servers lists even if at the end of a list.

Usage

- **Usage notes:**
  - OpenSSH for z/OS uses the session object token, SYSTOK-SESSION-ONLY, to exploit the ICSF PKCS #11 support.
  - This support applies to SSH protocol version 2 only.
  - **sshd** won't use ICSF to implement the arcfour, arcfour128 and arcfour256 ciphers when privilege separation is enabled.
  - **ssh** and **sshd** will fail if ICSF ciphers or MAC algorithms are required but ICSF isn't available.
  - ICSF ciphers and MAC algorithms are not supported when using the **ssh** -f option or the **ssh** ~& escape character.

Usage

- **FIPS 140-2 notes:**
  - ICSF PKCS #11 services can be configured to operate in compliance with FIPS 140-2 specifications. Refer to the ICSF FIPSMODE installation option.
  - OpenSSH for z/OS is still not considered a FIPS 140-2 compliant application. That is, it doesn't have a "FIPS 140-2 mode" of operation.

Usage

- **How the ICSF FIPSMODE installation option affects this support**
  - The following ICSF ciphers are supported when FIPS 140-2 compliance is required
    - aes128-cbc, aes192-cbc, aes256-cbc, 3des-cbc
    - rijndael-cbc@lysator.liu.se (same as aes256-cbc)
  - The following ICSF MAC algorithms are supported when FIPS 140-2 compliance is required
    - hmac-sha1, hmac-sha1-96
  - Other ICSF ciphers and MAC algorithms cannot be used (i.e. they will fail at run-time) when FIPS 140-2 compliance is required unless OpenSSH for z/OS is exempt.

Usage

- **Updates to the SMF Type 119 records for OpenSSH for z/OS**
  - Added new ICSF cipher values to the SMF_119SSH_Cipher field.
  - Added new ICSF MAC algorithm values to the SMF_119SSH_MAC field.
  - Updated the C-level mapping macros in /samples/ssh_smf.h
  - Updated the assembler mapping macros in SYS1.MACLIB(FOTSMF77)

Usage

- **When OpenSSH is setup to use ICSF to implement applicable ciphers and MAC algorithms, debug mode provides ICSF Query Algorithm (CSFIQA) debug statements to help determine how ICSF is implementing the ciphers and MAC algorithms. For example:**

```
debug2: ----------------------------------
debug2: CRYPTO    SIZE     KEY       SOURCE
debug2: ----------------------------------
debug2: AES       256      SECURE    COP
debug2: AES       256      SECURE    CPU
debug2: DES       56       SECURE    COP
debug2: DES       56       SECURE    CPU
debug2: MDC-2     128      NA        CPU
debug2: MDC-4     128      NA        CPU
debug2: MD5       128      NA        SW
debug2: RNGL      8192     NA        COP
debug2: RPMD-160  160      NA        SW
debug2: RSA-GEN   4096     SECURE    COP
debug2: RSA-KM    4096     SECURE    COP
debug2: RSA-SIG   4096     SECURE    COP
debug2: SHA-1     160      NA        CPU
debug2: SHA-2     512      NA        CPU
debug2: TDES      168      SECURE    COP
debug2: TDES      168      SECURE    CPU
```

Usage

- **To determine the cipher and MAC algorithm source used by OpenSSH for z/OS, start ssh in debug mode and look for debug statements like the following:**

```
debug1: mac_setup_by_id: hmac-sha1 from source ICSF
debug1: cipher_init: aes128-cbc from source ICSF.
```


Usage

- **Recommend using "any" option and example ciphers and MAC algorithms lists to start.**
- **ICSF "status" change during SSH sessions**
  - Off to On - Won't affect existing sessions, only future sessions.
  - On to Off - May cause existing sessions to fail.
- **Common issues with enablement:**
  - Problems with ICSF setup or configuration
  - Ciphers and MAC algorithms lists not modified appropriately

Agenda

- New OpenSSH for z/OS Extension
  - Overview
  - Packaging and installation
  - Usage

- **New: sudo for z/OS**
  - **>> Overview <<**
  - Packaging and installation
  - Usage
  - Examples
- Appendix

Overview

- **What is sudo?**

  sudo (su “do”) is an open source tool that allows a system administrator to delegate authority in order to give certain users (or groups of users) the ability to run some (or all) commands as a superuser or another user, while providing an audit trail of the commands and their arguments. It is a command-line UNIX application.

Overview

- **Problem Statement**
  - z/OS system administrators need a more granular and flexible method to minimize user privileges while still allowing users to get their work done.
- **Solution**
  - IBM ported the sudo open source tool to z/OS. sudo is commonly available on other UNIX/Linux platforms.
- **Benefits**
  - sudo for z/OS is designed to allow a z/OS system administrator to minimize user privileges while still allowing users to get their work done. **sudo for z/OS** is preferred over **su** since it doesn't require giving the invoking user "open" access to run as the target user.

SHARE in San Francisco 2013

Overview

- **Benefits (Continued)**
  - Today without sudo for z/OS, a z/OS system administrator could…
    (1) Allow users to share UID(0)
    (2) Allow users to be surrogates of a UID(0) user
    (i.e. **su –s <user>)**
    (3) Provide users with a UID(0) user's password (i.e. **su <user>**)
    (4) Give users BPX.SUPERUSER authority
    (i.e. **su** superuser mode)
    (5) Give users select UNIXPRIV authorities

    However, all of these options have inherent risks associated with them. They may provide a user with more privilege than the system administrator wants to provide. These risks result in the need for sudo for z/OS.

SHARE in San Francisco 2013

Overview

- **Benefits (Continued)**
  - sudo for z/OS doesn't require sharing UIDs or passwords, creating surrogates or granting excessive authority in order to allow users to get their work done. Additional customizations are possible, including the ability to have users run as non-UID(0) users.
  - sudo for z/OS has built-in logging of commands that are being run under the sudo authority.

SHARE in San Francisco 2013

Overview

- **Who maintains sudo and where can I find more information?**
  - IBM ported open source sudo version 1.7.2p2 to z/OS and modified the port for better z/OS integration.
  - Refer to http://www.ibm.com/systems/z/os/zos/features/unix/ported/suptlk/ for sudo for z/OS.
  - Refer to http://www.sudo.ws/ for open source sudo.

SHARE in San Francisco 2013

Agenda

- New OpenSSH for z/OS Extension
  - Overview
  - Packaging and installation
  - Usage

- **New: sudo for z/OS**
  - Overview
  - **>> Packaging and installation <<**
  - Usage
  - Examples
- Appendix

Packaging and installation

- **sudo for z/OS has been provided via APAR OA34949 (PTF UA59179) to IBM Ported Tools for z/OS: Supplementary Toolkit for z/OS (FMID HPUT110).**
  - sudo for z/OS is supported on z/OS 1.10 and later
  - z/OS 1.10 and z/OS 1.11 requirement: PTF for APAR OA32470 must be applied.
- **Incomplete HOLD ACTION in PTF for APAR OA34949 fix via PE APAR OA37129.**
- **See the "Installing Supplementary Toolkit for z/OS" section in the user's guide for details**

Packaging and installation

- **Pre-installation planning**
  - New directories must be created before installing the APAR.
  - sudo for z/OS requires a GID(0) group to be defined on your system.
  - Ensure file system contains enough available space.
  - Verify the z/OS release requirements noted on the previous slide.
- **See the "Pre-installation planning" section in the user's guide for details**

Packaging and installation

- **Post-installation setup and verification (Required)**
  - Enable sudo for z/OS - SAMPLIB member HPUTIFA provides an example
  - Copy the sudoers file to /etc/sudoers

```
# sudoers must have mode 0440 (i.e. read for owner and group).
# sudoers must be owned by UID(0) and GID(0).
cp -p /usr/lpp/ported/samples/sudoers /etc/sudoers
```

  - Customize the /etc/sudoers file for your installation using visudo

```
# By default, there's no sudo authority.
# By default, BPXROOT is the default runas and mailto user.
visudo
```

- **See the "Post-installation setup and verification" section in the user's guide for details**

Packaging and installation

- **Post-installation setup and verification (Recommended)**
  - Add a symbolic link to the man pages, if necessary
    ```
    /usr/man/C/man1/hpuza200.book
    # symlink --> /usr/lpp/ported/man/C/man1/hpuza200.book
    ```
  - Add a symbolic link to the message catalog
    ```
    /usr/lib/nls/msg/C/hpusudo.cat
    # symlink --> /usr/lpp/ported/lib/nls/msg/C/hpusudo.cat
    ```
  - Add a symbolic link to the binaries
    ```
    /usr/bin/sudo      # symlink --> /usr/lpp/ported/bin/sudo
    /usr/bin/sudoedit  # symlink --> /usr/lpp/ported/bin/sudoedit
    /usr/sbin/visudo   # symlink --> /usr/lpp/ported/bin/visudo
    ```
- **See the "Post-installation setup and verification" section in the user's guide for details**

Packaging and installation

- **Post-installation setup and verification (Recommended)**
  - Verify sudo for z/OS installation
    - sudo must be owned by UID(0)
    - sudo must have mode 4111 (i.e. execute for all and set-user-ID)
    - sudo must have noshareas extended attribute (i.e. extattr –s)
    - sudo must have the program control extended attribute (i.e. extattr +p)
- **See the "Post-installation setup and verification" section in the user's guide for details**

Packaging and installation

- **Updated toolkit parts for sudo for z/OS**
  - /usr/lpp/ported/Ported_Tools_License.readme
  - /usr/lpp/ported/man/C/man1/hpuza200.book
  - SYS1.SAMPLIB(HPUTIFA)
  - SYS1.SAMPLIB(HPUTMKDR)
- **New sudo for z/OS parts**
  - /usr/lpp/ported/bin/base/sudo-1.7.2p2
  - /usr/lpp/ported/bin/base/visudo-1.7.2p2
  - /usr/lpp/ported/samples/sudoers
  - /usr/lpp/ported/lib/nls/msg/C/hpusudo.cat
    # Supporting directories (lib/nls/msg/C) are also new.

SHARE in San Francisco 2013

Packaging and installation

- **New sudo for z/OS symbolic links**
  - /usr/lpp/ported/bin/sudo # --> base/sudo-1.7.2p2
  - /usr/lpp/ported/bin/sudoedit # --> base/sudoedit-1.7.2p2
  - /usr/lpp/ported/bin/visudo # --> base/visudo-1.7.2p2
- **New sudo for z/OS hard links**
  - /usr/lpp/ported/bin/base/sudoedit-1.7.2p2 # --> ./sudo-1.7.2p2
  - /usr/lpp/ported/IBM/HPUDXSUD # --> ../bin/base/sudo-1.7.2p2
  - /usr/lpp/ported/IBM/HPUDXVIS # --> ../bin/base/visudo-1.7.2p2
  - /usr/lpp/ported/IBM/HPUDUERS # --> ../samples/sudoers
  - /usr/lpp/ported/IBM/HPUDRCAT # --> ../lib/nls/msg/C/hpusudo.cat

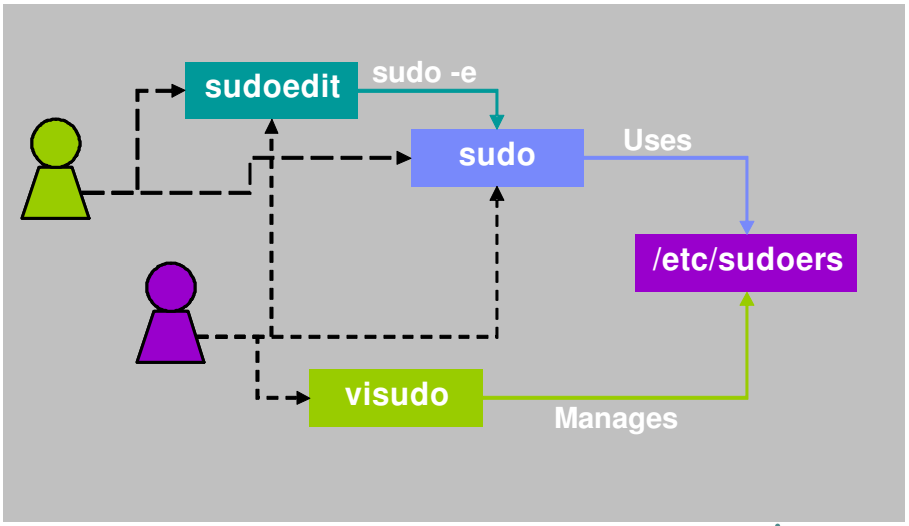
SHARE in San Francisco 2013

Agenda

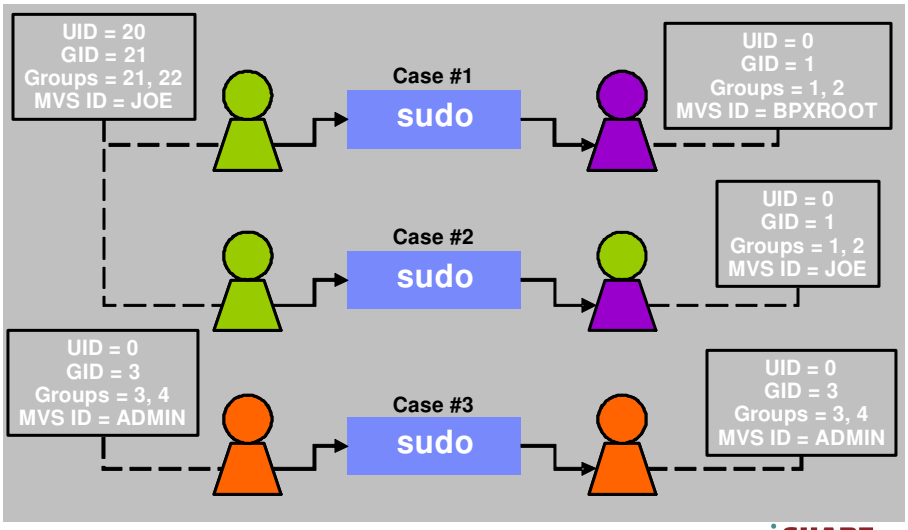
- New OpenSSH for z/OS Extension
  - Overview
  - Packaging and installation
  - Usage

- **New: sudo for z/OS**
  - Overview
  - Packaging and installation
  - **>> Usage <<**
  - Examples
- Appendix

Usage



Usage



Usage

Command / Authority	z/OS UNIX ID Change	MVS ID Change	Shell Access	Command Control
sudo	Optional	Optional	Optional	Yes
su <user>	Yes	Yes	Yes	No
su –s <user> (i.e. SURROGAT)	Yes	Yes	Yes	No
su (i.e. BPX.SUPERUSER)	Yes	No	Yes	No
UNIXPRIV	No	No	No	Partial

Usage

- **/etc/sudoers security**
  - Contains sensitive information and must be protected.
  - Must be owned by UID(0) and GID(0) and permissions must be 0440. If not, sudo, sudoedit or visudo will either fix the ownership and permissions or will fail.
  - Only superusers should edit the file (ideally using visudo).
  - Only superusers should be members of the GID(0) group.

Usage

- **/etc/sudoers example**

```
# The sudoers file is composed of two types of entries: aliases (basically
# variables) and user specifications (which specify who may run what).

# The following Defaults specifications are either unique to z/OS or
# have had their default values changed for z/OS.
Defaults !path_info
Defaults ignore_dot
Defaults zos_set_mvs_identity=never

# For better control of which user is selected as the default user,
# the runas_default specification should be set to your desired user.
# You should also set the mailto specification to the desired user.
# The default for both is BPXROOT.
Defaults runas_default=BPXROOT
Defaults mailto=BPXROOT
```

Usage

- /etc/sudoers example (continued)

```
# GOOD user specifications:

# Sample user privilege specification to allow a non-UID(0) user (rtheis)
# to edit a UID(0) owned file to which rtheis doesn't have permission.
rtheis ALL= sudoedit /u/root/sharedFiles/file1

# Allow rtheis to display open files for a file system.
rtheis ALL= /bin/fuser -c /tst

# Allow rtheis to run a certain shell script with no arguments.
rtheis ALL= /u/root/script.sh ""
```


SHARE in San Francisco 2013

Usage

- /etc/sudoers example (continued)

```
# DANGEROUS user specifications:

# Allows shell escapes
# rtheis ALL= /bin/vi /u/root/sharedFiles/file1

# Allows shell access
# rtheis ALL= /bin/sh, /bin/tcsh

# Allows identity chaining
# rtheis ALL= /bin/su, /bin/sudo
```


SHARE in San Francisco 2013

Usage

- **Security recommendations for user specifications**
  - sudoers grammar (EBNF) can be confusing so use examples
  - Make user specifications as specific as possible
  - Minimize use of the ALL alias and sudo “chaining”
  - Specify commands with arguments or use “” to ensure commands are run without arguments
  - Subtracting commands from the ALL alias using the ‘!’ operator is generally not effective
  - Minimize shell access and shell escapes
- **Suggest reading the user’s guide before using sudo**
  - Specific references: “Preventing shell escapes”, “Security notes” for sudo and “Security notes” for sudoers

Usage

- **Default option value differences between z/OS and open source**
  - sudoers **ignore_dot** option:
    z/OS default = “on”
    open source default = “off”
  - sudoers **runas_default** and **mailto** options:
    z/OS default = “BPXROOT”
    open source default = “root”
  - sudoers **path_info** option:
    z/OS default = “off”
    open source default = “on”

Usage

- **Unsupported open source functionality on z/OS**
  - sudo options: -A askpass, -a type, -c class, -r role, -t type
  - sudoers options: askpass, ignore_local_sudoers, insults, long_opt_prompt, noexec, noexec_file, passprompt_override, pwfeedback, role, rootpw, stay_setuid, sudoers_locale, type, use_loginclass, visiblepw
  - sudoers specifications: netgroup, nonunixgroup, NOEXEC / EXEC

Usage

- **New z/OS-specific functionality**
  - Environment variables: _ZOS_SUDO_NOMSGID and _ZOS_SUDO_DEBUG
  - sudoers options: zos_set_mvs_identity
  - sudoers specifications: ZOS_SET_MVS_IDENTITY / NO_ZOS_SET_MVS_IDENTITY

Caution: Allowing **sudo for z/OS** to change MVS identity means that **su** can be used to do the same!

Agenda

- New OpenSSH for z/OS Extension
  - Overview
  - Packaging and installation
  - Usage

- **New: sudo for z/OS**
  - Overview
  - Packaging and installation
  - Usage
  - **>> Examples <<**
- Appendix

Examples

- **Example #1:** Allow users on a team (BACKUPS) the ability to run a specific pax command as a specific UID(0) administrator (admin) with specific arguments determined by the administrator.
  - **/etc/sudoers file entries:**

```
Defaults umask=077
User_Alias BACKUPS = june, fred, mary
BACKUPS ALL = (admin) /bin/pax -x pax -wf /u/code/src.pax /u/code/src
```

  - **sudo command allowed:**

```
sudo –u admin pax -x pax -wf /u/code/src.pax /u/code/src
```

  - **Benefits to using sudo:**
    - Backup team not allowed to view the data they pax'd as an administrator.
    - Backup team not allowed to run other pax commands or change pax options as an administrator.
    - Audit trail provided for every backup done by the backup team.

Examples

- **Example #2:** Log all commands run by UID(0) user.
  - **/etc/sudoers file entries:**
    ```
    admin ALL=(admin) ALL
    ```
  - **sudo command allowed:**
    ```
    sudo rm –rf /u/baduser
    ```
  - **Benefits to using sudo:**
    - Audit trail provided for every command run via sudo by UID(0) user admin.
  - **Example syslog audit entry created by sudo:**
    ```
    Aug 25 08:00:04 SY1 sudo:   admin : TTY=ttyp0000 ;
    PWD=/SYSTEM/tmp/syslogd ; USER=admin ; GROUP=admingrp ;
    COMMAND=/bin/rm –rf /u/baduser
    ```


Questions?

Agenda

- New OpenSSH for z/OS Extension
  - Overview
  - Packaging and installation
  - Usage

- New: sudo for z/OS
  - Overview
  - Packaging and installation
  - Usage
  - Examples
- **>> Appendix <<**

Appendix

- **See the updated "IBM Ported Tools for z/OS: OpenSSH User's Guide" for more information**
  (Order Number: SA23-2246-01)
- **Website References**
  - IBM Ported Tools for z/OS: http://www.ibm.com/systems/z/os/zos/features/unix/ported/
  - IBM Ported Tools for z/OS: OpenSSH: http://www.ibm.com/systems/z/os/zos/features/unix/ported/openssh/
  - OpenSSH: http://www.openssh.org/
  - OpenSSL: http://www.openssl.org/

Appendix

- **ICSF Reference Guides:**
  - z/OS Cryptographic Services ICSF Overview (Order Number: SA22-7519-13)
  - z/OS Cryptographic Services ICSF Administrator's Guide (Order Number: SA22-7521-14)
  - z/OS Cryptographic Services ICSF System Programmer's Guide (Order Number: SA22-7520-14)
  - z/OS Cryptographic Services ICSF Application Programmer's Guide (Order Number: SA22-7522-13)
  - z/OS Cryptographic Services Writing PKCS #11 Applications (Order Number: SA23-2231-02)
- **Other Reference Guides:**
  - Program Directory for IBM Ported Tools for z/OS (Order Number: GI10-0769-06)

Appendix

- **See the updated "IBM Ported Tools for z/OS: Supplementary Toolkit for z/OS Feature User's Guide" for more details on sudo for z/OS.** (Order Number: SA23-2234)
- **Website References**
  - IBM Ported Tools for z/OS: http://www.ibm.com/systems/z/os/zos/features/unix/ported/
  - IBM Ported Tools for z/OS: Supplementary Toolkit for z/OS: http://www.ibm.com/systems/z/os/zos/features/unix/ported/suptlk/
  - sudo: http://www.sudo.ws/

System z Social Media

- System z official Twitter handle:
  - **@ibm_system_z**
- Top Facebook pages related to System z:
  - **Systemz Mainframe**
  - **IBM System z on Campus**
  - **IBM Mainframe Professionals**
  - **Millennial Mainframer**
- Top LinkedIn Groups related to System z:
  - **Mainframe Experts Network**
  - **Mainframe**
  - **IBM Mainframe**
  - **System z Advocates**
  - **Cloud Mainframe Computing**
- YouTube
  - **IBM System z**

- Leading Blogs related to System z:
  - **Evangelizing Mainframe (Destination z blog)**
  - **Mainframe Performance Topics**
  - **Common Sense**
  - **Enterprise Class Innovation: System z perspectives**
  - **Mainframe**
  - **MainframeZone**
  - **Smarter Computing Blog**
  - **Millennial Mainframer**