#SHAREorg

SHARE
Technology · Connections · Results

New OpenSSH for z/OS Extension

Richard Theis (rtheis@us.ibm.com)
IBM Rochester, MN

March 15, 2012
Session 10865

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

>> Overview <<

Packaging and installation

Usage

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.

Agenda

Overview

>> Packaging and installation <<

Usage

Appendix

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.

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)

Agenda

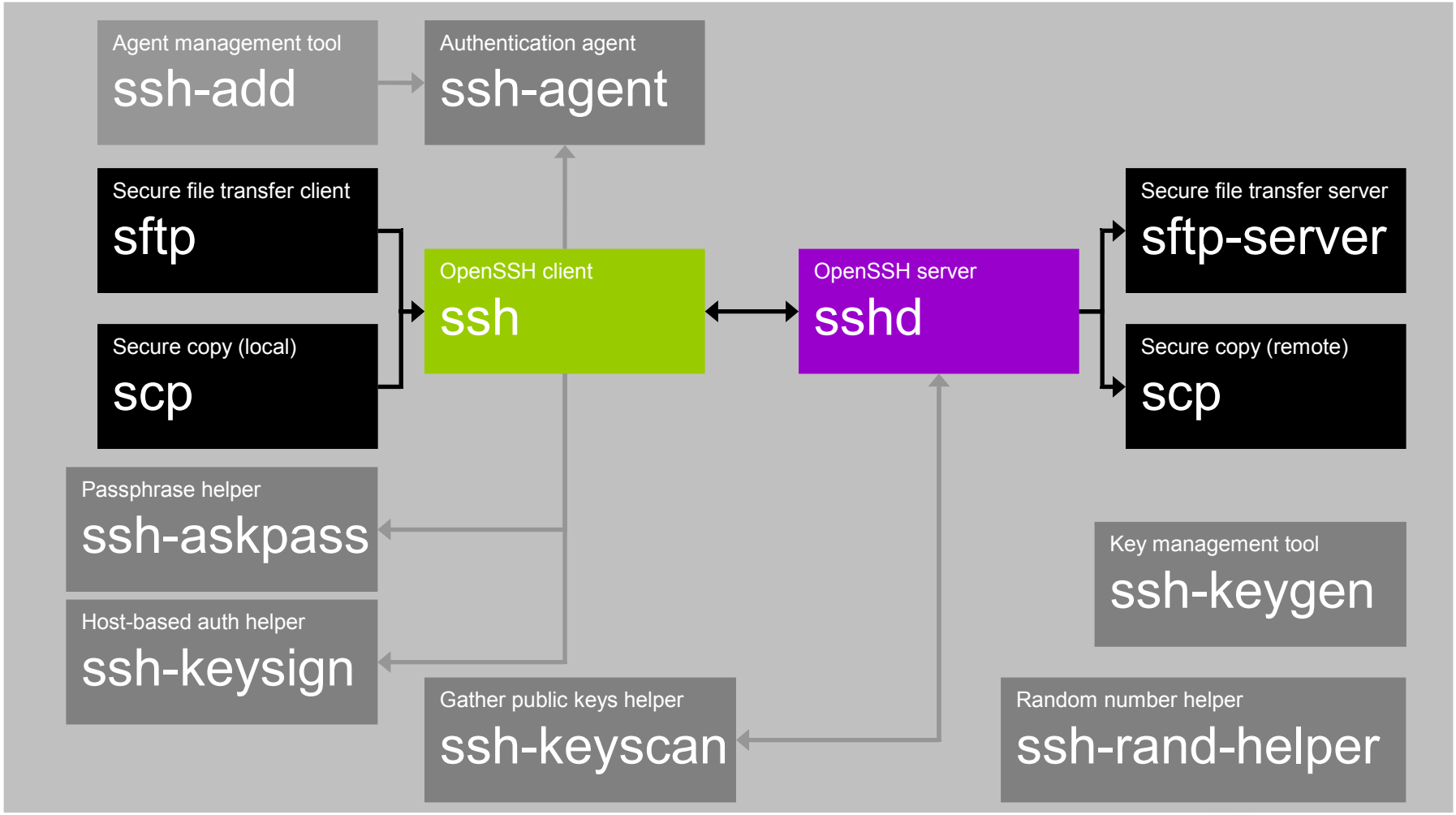
Overview

Packaging and installation

>> Usage <<

Appendix

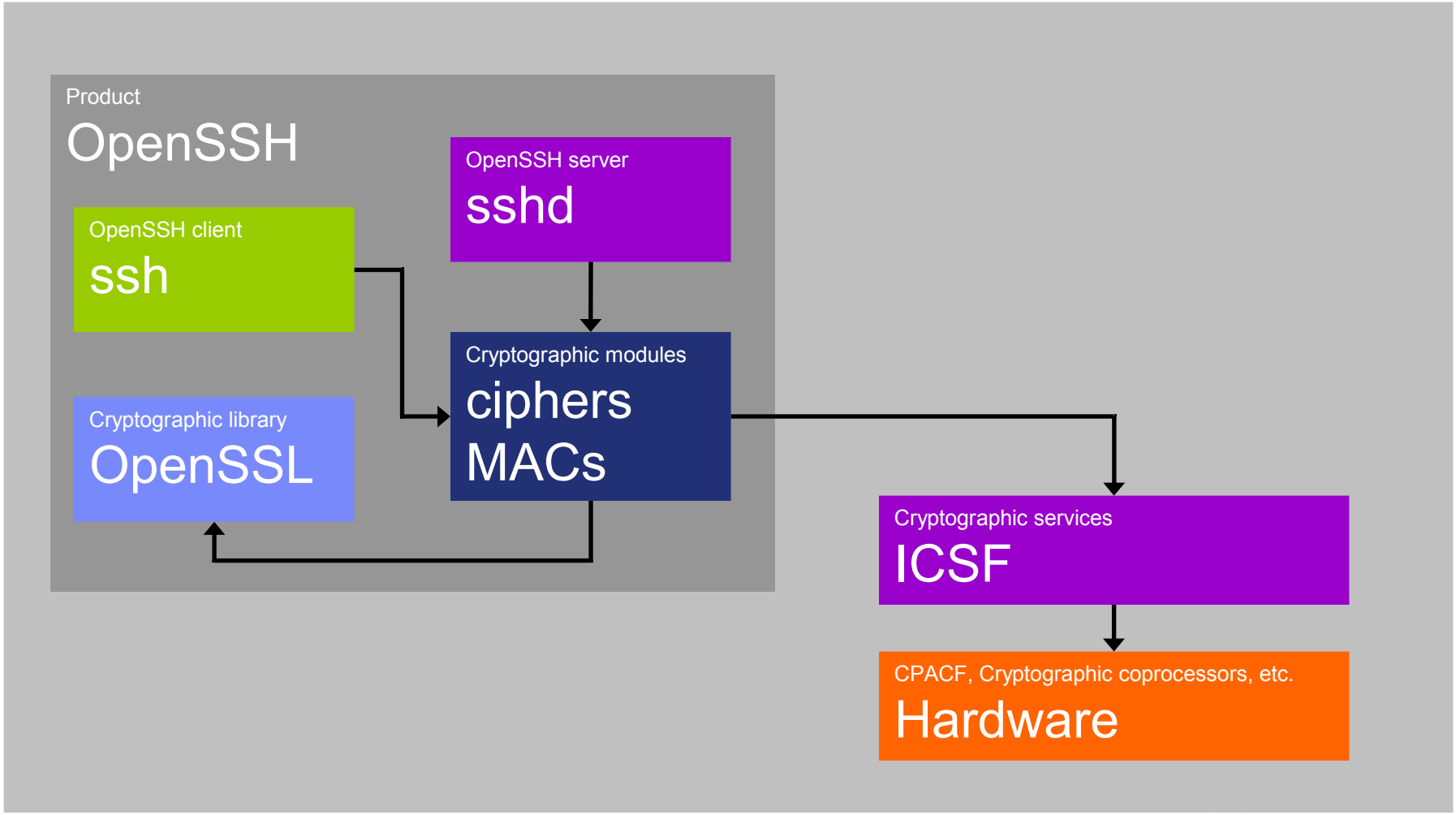
Usage



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


Product
OpenSSH
OpenSSH server
sshd
OpenSSH client
ssh
Cryptographic modules
ciphers
MACs
Cryptographic library
OpenSSL
Cryptographic services
ICSF
CPACF, Cryptographic coprocessors, etc.
Hardware

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.

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.

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

Overview

Packaging and installation

Usage

>> 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)