OpenSSH for z/OS: New Features and Functions

Stephen Goetze
Kirk Wolf
Dovetailed Technologies, LLC

#SHAREorg

SHARE is an independent volunteer-run information technology association that provides **education**, professional **networking** and industry **influence**.

Trademarks

- Co:Z® is a registered trademark of Dovetailed Technologies, LLC
- z/OS® is a registered trademark of IBM Corporation
- Windows, PowerShell are trademarks of Microsoft Corporation

Dovetailed Technologies

We provide z/OS customers world wide with innovative solutions that enhance and transform traditional mainframe workloads:

- Co:Z Co-Processing Toolkit for z/OS
  - z/OS Enabled SFTP, z/OS Hybrid Batch, z/OS Unix Batch integration
  - Uses IBM z/OS OpenSSH or Ported Tools OpenSSH

- JZOS

  acquired by IBM in and now part of the z/OS Java SDK

Agenda

- What is SSH?
- Review 2015 major releases:
  - IBM Ported Tools for z/OS V1.3 – OpenSSH
  - z/OS V2R2 OpenSSH
- Migration considerations

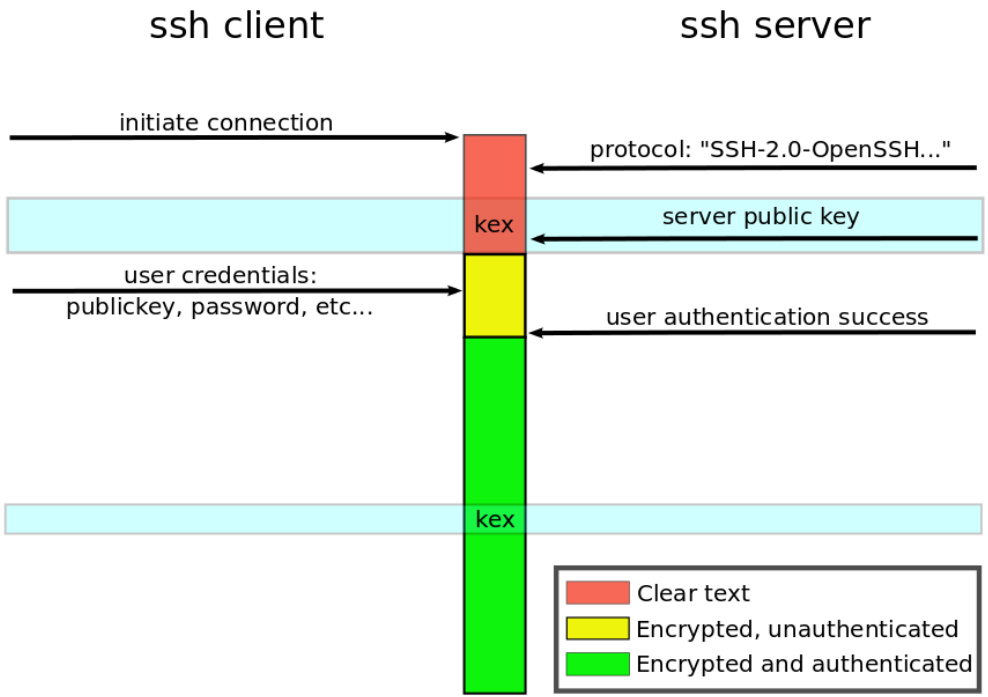
What is SSH?

- The IETF SSH-2 standard protocol (RFC 4251 etc)
- Features:
  - A secure (encrypted) connection over **one** TCP/IP socket between a client and a server
  - Authentication of both user and host
  - (optional) LZ compression
  - Support for one or more simultaneous ***application channels*** over the same connection: terminal, sftp, command, port forwarding, ...
- There are many compatible implementations
  - **OpenSSH** is by far the most popular; it is a default package on all Unix/Linux distributions. ***Soon available from Microsoft for Windows and PowerShell***.
  - PuTTY is a popular free Windows client

SSH2 Crypto at-a-glance

- **Key Exchange – “kex”**
  - Some variant of the Diffie-Hellman algorithm
  - Client generated random number and the server key are used to:
    - Allow the client to authenticate the identity of the server
    - Cooperatively generate and exchange a secret session key
    - Supports session rekeying. Typically once/hour or GB.
- **User Authentication**
  - At start of session, a password or **user** key pair can be used to authenticate the user to the server.
- **Session Encryption**
  - A symmetric Cipher uses the shared session key to encrypt the packet payload.
  - A MAC algorithm (typically SHA-1) is used to generate a hash of each packet.

SSH Encryption and Authentication



IBM z/OS OpenSSH – By the Numbers

- IBM Ported Tools or z/OS – OpenSSH
  - V1.2.0 – HOS1120 – z/OS 1.10–2.1 - OpenSSH 5.0p1 (July ‘10)
  - ***V1.3.0 – HOS1130 – z/OS 1.13–2.1 - OpenSSH 6.4p1 (Jan ‘15)***
- IBM z/OS V2R2 OpenSSH
  - V2.2.0 – HOS2220 – z/OS 2.2 - OpenSSH 6.4p1 (the same code as HOS1130)
  - ***V2.2.0 - HOS2220+UA79909 - z/OS 2.2 - OpenSSH 6.4p1 (Dec‘15)***

IBM Ported Tools for z/OS V1.3 – OpenSSH

Summary of Changes

- Base upgraded from OpenSSH 5.0p1 to 6.4p1
- ICSF acceleration of CTR mode AES ciphers
  - CTR mode is now preferred over CBC*
- New SMF 119 record types and detail
- Enabled ssh client to be invoked under TSO/OMVS shell
  - entry of password credentials not permitted
- Relaxed syntax of IdentityKeyRingLabel
  - double quotes optional when entered from ssh, sftp, or scp command line
- Remains a no-charge z/OS product; normal IBM support

*http://www.kb.cert.org/vuls/id/958563

IBM z/OS V2R2 OpenSSH

Summary of changes

- FMID HOS2220
- Now included as a base feature in z/OS V2R2 (June 2015)
- The ***same*** as HOS1130 (Ported Tools OpenSSH V1.3.0)

IBM z/OS V2R2 OpenSSH - PTF UA79909

Summary of changes (available: December 2015)

- z/OS OpenSSH can be configured to run in FIPS 140-2 mode
- Support for GSSAPI/Kerberos authentication
  - Single sign-on interoperability with other Kerberos systems, including Windows Active Directory.
- zEDC enablement of zlib compression feature
- Can enable/disable ASCII-EBCDIC conversion by channel type
- Enhanced SOCKS proxy support

➢ Satisfies Statement of Direction: ZP15-0006, January 14, 2015

IBM Ported Tools for z/OS V1.3 – OpenSSH

Product Notes

- New release (HOS1130) installs over the previous release (HOS1120)
- HOS1130 is supported on z/OS V1R13 – V2R1
- HOS1120 supported through z/OS V2R1, but withdrawn from marketing at HOS1130 GA.
- /dev/random is now **required** – HOS1130 will not run without ICSF active!
- Verifying version

```
$ ssh -V
OpenSSH_6.4p1, OpenSSL 1.0.1c 10 May 2012

$ /usr/sbin/sshd -d -t
…
debug1: sshd version OpenSSH_6.4p1, OpenSSL 1.0.1c 10 May 2012
```

HOS1130: OpenSSH 6.4p1 new features

- Key Exchange algorithms can now be specified (-oKexAlgorithms)

  diffie-hellman-group1-sha1,
  diffie-hellman-group14-sha1,
  diffie-hellman-group-exchange-sha1,
  diffie-hellman-group-exchange-sha256,
  ecdh-sha2-nistp256,
  ecdh-sha2-nistp384,
  ecdh-sha2-nistp521

- NIST Elliptic-curve algorithms added

Note: new algorithms **highlighted**

HOS1130: OpenSSH 6.4p1 new features

- Key Algorithms – used for ssh host (server) or user keys

  ssh-rsa,ssh-dss,
  ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521,
  ssh-rsa-cert-v01@openssh.com,
  ssh-dss-cert-v01@openssh.com,
  ecdsa-sha2-nistp256-cert-v01@openssh.com,
  ecdsa-sha2-nistp384-cert-v01@openssh.com,
  ecdsa-sha2-nistp521-cert-v01@openssh.com,
  ssh-rsa-cert-v00@openssh.com,
  ssh-dss-cert-v00@openssh.com

- NIST Elliptic-curve DSA w/ SHA-2 algorithms added
- OpenSSH “certificates” added (more later)

Notes: new algorithms **highlighted ,** non-standard non-RFC names have “@openssh.com”

HOS1130: OpenSSH 6.4p1 new features

- Cipher algorithms – default preference order shown

  aes128-ctr**,aes192-ctr**,aes256-ctr**,arcfour256,arcfour128,
  aes128-gcm@openssh.com,aes256-gcm@openssh.com,
  aes128-cbc*,3des-cbc*,blowfish-cbc,cast128-cbc,aes192-cbc*,
  aes256-cbc*,arcfour,rijndael-cbc@lysator.liu.se*

- AES GCM (Galois/Counter Mode) ciphers added to OpenSSH
  - Function as both Cipher and HMAC in one
- AES CTR mode ICSF support has been added to HOS1130
  - Accelerates the most widely used OpenSSH Ciphers

Note: new algorithms **highlighted,** ICSF support noted with "*" or "**"(new)

HOS1130: OpenSSH 6.4p1 new features

- MAC algorithms – default preference order shown

  hmac-md5-etm@openssh.com*,hmac-sha1-etm@openssh.com*,
  umac-64-etm@openssh.com,umac-128-etm@openssh.com,
  hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com**,**
  hmac-ripemd160-etm@openssh.com*,
  hmac-sha1-96-etm@openssh.com*,hmac-md5-96-etm@openssh.com*,
  hmac-md5*,hmac-sha1*,
  umac-64@openssh.com,umac-128@openssh.com,hmac-sha2-256,hmac-sha2-512**,**
  hmac-ripemd160*,hmac-ripemd160@openssh.com*,hmac-sha1-96*,hmac-md5-96*

  – SHA-2 algorithm added (with ICSF support)
  – UMAC algorithm support added
  – "-etm@openssh.com" algorithms are **not** new algorithms!

  They are variants that indicate that the MAC is calculated **after** encryption ("Encrypt-then-MAC") rather than the other way around. The community now considers this more secure (in theory).

Note: new algorithms **highlighted,** ICSF support noted with "*" or "**"(new)

HOS1130: OpenSSH 6.4p1 new features

- Dynamic port assignment for remote port forwarding
  - ssh -R 0:host:port
  - A remote port of “0” can be specified in which case a dynamic port will be assigned on the server.
  - The client will report a message with the specific ephemeral port assigned.
- More flexibility in configuration files
  - Match blocks have more criteria and can include more options within the block.
- Support for public key (user and host) certificates
  - These are not X.509 certificates, but a simpler implementation that is unique to OpenSSH.
    - A single key (“CA key”) may sign (vouch for) the public keys of many users or servers. If a host or user trusts the CA public key, then it implicitly accepts the keys that have been signed by it.
  - For more information, see the User's Guide / man page for the ssh-keygen command.
  - These have been available for a few years, but are not widely used.
  - No z/OS Key Ring support for these or their associated keys.

HOS1130: OpenSSH 6.4p1 new features

- Multi factor authentication methods
  - The server may specify that more than one authentication method is required for a/all user(s). For example:
  - AuthenticationMethods publickey,password publickey,publickey

- SFTP enhancements
  - Support for recursive file transfer in a directory tree (get/put -r)
  - sftp server read-only mode
  - sftp “df” command - displays file system attributes
  - improved performance of directory listings
  - “ls -h” option - human readable file attribute units
  - ***No*** sftp support for MVS datasets, spool files, etc.

HOS1130: New SMF type 119 records

- Two new SMF 119 records were added:
  - type 94(x'5D') Client connection started record
  - type 95(x'5E') Server connection started record
- If SMF recording is configured, these records will be written just after the user has been authenticated by the server.
  - in zos_ssh_config / zos_sshd_config
- The content of these records is identical, and a subset of other 119 SSH records:
  - standard SMF 119 header
  - common 119 TCP/IP identification section
  - SSH common security section (identifies which algorithm(s) were used)
- BPX.SMF permission is now required for ssh client users if SMF recording is enabled, since the **ssh** command is not APF authorized.
- C-level mapping macros in /samples/ssh_smf.h and the assembler mapping macros in SYS1.MACLIB(FOTSMF77) have been updated.

HOS1130: Migration Considerations: Config Files

- Customers with prior releases should review their configuration files to determine applicability of new features. Many new configuration options have been added through OpenSSH 6.4, and defaults for others have been changed.
  - ssh_config
  - sshd_config
  - zos_ssh_config
  - zos_sshd_config
- As in previous releases, protocol 1 is disabled by default.
- RhostsAuthenticaion (protocol 1 only) was removed in OpenSSH 3.7 and is no longer supported. RhostsRSAAuthentication may be used as a more secure alternative.

HOS1130: Migration Considerations: SFTP

- OpenSSH 6.4 changes sftp so that non-error messages are not printed to stdout if running a batch file (-b).
  - In effect, the -q (quiet mode) option is turned on with -b and cannot be turned off.
  - Since this will impact many customers, it has been changed in HOS1130 so that -b does not force -q.
  - The -q option can be specified in addition to -b. Therefore this is not actually a migration action, but the behavior will not be consistent with other implementations.

HOS1130: Migration Considerations: /dev/random

- OpenSSH 6.4 no longer supports the use of ssh-rand-helper
- In HOS1130, neither the ssh client or sshd server will run unless the UNIX /dev/random device is working.
  - ICSF support of /dev/random is now **REQUIRED**.
    - Version HCR7780 or later must be installed and running
    - With HCR77A0, a **crypto card is NOT required**!
    - With HCR77A1, CSFRNG check can be skipped by defining resource `CSF.CSFSERV.AUTH.CSFRNG.DISABLE` in class `XFACILIT`
  - If `/dev/random` is not available, then ssh/sshd will fail with:

```
FOTS1949 PRNG is not seeded.  Please activate the Integrated Cryptographic Service
Facility (ICSF)
```

HOS1130: Configuration and Tuning

- Our guide for setup and tuning z/OS OpenSSH:

  IBM Ported Tools OpenSSH 1.3 / z/OS V2R2 OpenSSH - Quick Install Guide
  http://dovetail.com/docs/pt-quick-inst/index.html

  – Common install customization path
  – ICSF (for /dev/random, Ciphers, and MACS)
  – LE tuning
  – etc.

- For more information, see also:

  IBM Ported Tools for z/OS: OpenSSH
  http://www.ibm.com/systems/z/os/zos/features/unix/ported/openssh/

IBM z/OS V2R2 OpenSSH - PTF UA79909

Review: Summary of changes

HOS2220 PTF UA79909 (available December 2015)

- z/OS OpenSSH can be configured to run in FIPS 140-2 mode
- Support for GSSAPI/Kerberos authentication
  - Single sign-on interoperability with other Kerberos systems, including Windows Active Directory.
- zEDC enablement of zlib compression feature
- Enhanced SOCKS proxy support
- Can enable/disable ASCII-EBCDIC conversion by channel type

- z/OS V2R2 OpenSSH – User's Guide
  http://publibz.boulder.ibm.com/epubs/pdf/fot1zo01.pdf

UA79909: What is FIPS 140-2?

- What is FIPS 140-2?
  - US Government standard for IT systems that process sensitive data.
  - Crypto operations must be done in certified “cryptographic modules”
  - Restricted set of crypto algorithms and operations
  - Cryptographic Module Certification Program jointly operated by NIST (US) and CSEC (Canada)
  - A good overview:
    “Understanding FIPS 140-2 and z/OS Systems SSL”
    ftp://public.dhe.ibm.com/s390/zos/racf/pdf/share_2013_02_understanding_fips_and_zos_system_ssl.pdf

UA79909: z/OS OpenSSH FIPSMode

- Enabled by “FIPSMode=yes” ssh and sshd option
  - on command line, zos_ssh_config, or zos_sshd_config
- When enabled:
  - Only FIPS compliant crypto algorithms (operations) are allowed
  - All crypto operations are done using ICSF and System SSL
  - All keys must be stored in Key Rings (files not supported)
- Prerequisites
  - ICSF configured to support FIPS
  - CiphersSource/MACsSource/**KexAlgorithmsSource** = ICSF

UA79909: z/OS OpenSSH FIPSMode

- Prerequisites (continued)
  - **Ciphers** option: algorithms must be limited to:
    aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,aes192-cbc,aes256-cbc,
    3des-cbc
  - **MACs** option: algorithms must be limited to:
    hmac-sha1,hmac-sha2-256,hmac-sha2-512,hmac-sha1-96,
    hmac-sha1-etm@openssh.com,hmac-sha2-256-etm@openssh.com,
    hmac-sha2-512-etm@openssh.com,hmac-sha1-96-etm@openssh.com,
  - **KeyAlgorithms** option: algorithms must be limited to:
    ssh-rsa, ssh-dss
  - **KexAlgorithms** option: algorithms must be limited to:
    diffie-hellman-group1-sha1,diffie-hellman-group14-sha1,
    diffie-hellman-group-exchange-sha1,diffie-hellman-group-exchange-
    sha256,ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521

UA79909: GSSAPI (Kerberos) support

- z/OS OpenSSH now supports the GSSAPI options from the base open source OpenSSH.
- Includes GSSAPIKeyExchange patches that are included in most Unix/Linux distributions (eliminates rqt. for “known_hosts”)
- Provides for “single sign-on” interoperability with Kerberos enabled clients and servers, included Windows Active Directory.

➢ **Session #18221 “The Three Headed Dog Ate My SSH Keys!”** (following this session in this room)

➢ **“Using OpenSSH in a Single Sign-on Corporate Environment with z/OS, Windows, and Linux”** http://dovetail.com/docs/ssh/kereros_sso.pdf

UA79909: zEDC-enabled zlib compression

- z/OS OpenSSH now uses IBM supplied zlib, which will exploit zEDC card(s) if present.
- option: zEDCCompression=yes
  - Specify on: command line, zos_sshd_config, zos_sshd_config
  - Should **only** specify for sftp/scp sessions.

UA79909: Improved SOCKS proxy support

- z/OS OpenSSH had previously supported the **ProxyCommand** option, which allows an external command to create a proxy connection.
- UA79909 adds **ProxyUseFdPass** option (from OpenSSH 6.5p1), which allows the external command to navigate the SOCKS proxy and then pass the connection back to ssh and terminate, which eliminates the ongoing overhead of piping the connection through a separate command.
- z/OS does **not** supply a SOCKS proxy client command, but one is available as a free download at dovetail.com

```
# example ssh_config
Host *.somedomain.com
ProxyCommand ssh-proxyc -v -E -p 192.168.1.105:9989 %h %p
ProxyUseFDPass yes
```

UA79909: new ChannelConvert option

- Previously, z/OS OpenSSH conversion is fixed:
  - Convert ASCII<->EBDIC: “shell”, “exec”
  - Binary: “subsystem”, “direct-tcpip”, “forwarded-tcpip”, “stdio-forward”
- The new “ChannelConvert” option can be used to change the default (except for “shell”)
  - Default: ChannelConvert shell,exec
  - Example:

```
ssh -oChannelConvert=shell user@linux1 cat my.tar | pax -rv
```

References

- IBM Ported Tools for z/OS 1.3.0: OpenSSH User's Guide
  (Order Number: SA23-2246-***03***)
- IBM z/OS V2R2 OpenSSH: User's Guide
  (Order number: SC27-6806-01)
- Dovetailed Technologies Resources (dovetail.com)
  - IBM Ported Tools OpenSSH / z/OS V2R2 OpenSSH
    - Quick Install Guide
    http://dovetail.com/docs/pt-quick-inst/index.html
  - Using OpenSSH in a Single Sign-on Corporate Environment with z/OS, Windows, and Linux
    http://dovetail.com/docs/ssh/kereros_sso.pdf
  - Dovetail webinar recordings:
    - IBM Ported Tools OpenSSH – Key Authentication
    - IBM Ported Tools OpenSSH – Using Key Rings
  - SSH proxy client. See "ssh-proxyc" at: http://dovetail.com/downloads/coz

- Website References:
  - IBM Ported Tools for z/OS - OpenSSH
    http://www.ibm.com/servers/eserver/zseries/zos/unix/ported/openssh/
  - OpenSSH http://www.openssh.org/

References

- ICSF Reference Guides:
  - z/OS Cryptographic Services ICSF Overview (Order Number: SA22-7519)
  - z/OS Cryptographic Services ICSF Administrator's Guide (Order Number: SA22-7521)
  - z/OS Cryptographic Services ICSF System Programmer's Guide (Order Number: SA22-7520)
  - z/OS Cryptographic Services ICSF Application Programmer's Guide (Order Number: SA22-7522)
  - z/OS Cryptographic Services Writing PKCS #11 Applications (Order Number: SA23-2231)
  - ftp://public.dhe.ibm.com/s390/zos/icsf/pdf/OA45548.pdf
- Other References:
  - Program Directory for IBM Ported Tools for z/OS (V1.3.0) (Order Number: GI10-0769)
  - Understanding FIPS 140-2 and z/OS Systems SSL ftp://public.dhe.ibm.com/s390/zos/racf/pdf/share_2013_02_understanding_fips_and_zos_system_ssl.pdf

Appendix: HOS1130 Configuration and Tuning

- The following information is taken from:

  IBM Ported Tools OpenSSH / z/OS V2R2 OpenSSH - Quick Install Guide
  http://dovetail.com/docs/pt-quick-inst/index.html

- For more information, see also:

  IBM Ported Tools for z/OS: OpenSSH
  http://www.ibm.com/servers/eserver/zseries/zos/unix/ported/openssh/index.html

Using ICSF to enable /dev/random

- Required for HOS1130
- Need to allow required users access to ICSF CSFRNG service. For most environments, this can be granted to all:

```
RDEFINE CSFSERV CSFRNG UACC(NONE)
PERMIT CSFRNG CLASS(CSFSERV) ID(*) ACCESS(READ)
SETROPTS RACLIST(CSFSERV) REFRESH
```

- You must authorize all userids that use ssh including both **sshd** userids.
- **Note:** With HCR77A1, this can be skipped by defining resource `CSF.CSFSERV.AUTH.CSFRNG.DISABLE` in class `XFACILIT`

To test (from a normal z/OS user UNIX shell):

```
$ head /dev/random | od -x
```

ICSF Cipher and MAC Acceleration

- ICSF must be active
- CPACF - processor feature 3863
  - free and enabled by default in most countries
- Properly configured, ICSF and CPACF instructions can reduce overall CPU usage by > 50%.
- PTF for APAR OA45548 must be installed to take advantage of AES-CTR mode.

ICSF Cipher and MAC Acceleration

- The following CSFSERV profiles control access:
  - CSFIQA - ICSF Query Algorithm
  - CSF1TRC - PKCS #11 Token record create
  - CSF1TRD - PKCS #11 Token record delete
  - CSF1SKE - PKCS #11 Secret key encrypt
  - CSF1SKD - PKCS #11 Secret key decrypt
  - CSFOWH - One-Way Hash Generate

ICSF Cipher and MAC Acceleration

```
RDEFINE CSFIQA  CLASS(CSFSERV) UACC(NONE)
RDEFINE CSF1TRC CLASS(CSFSERV) UACC(NONE)
RDEFINE CSF1TRD CLASS(CSFSERV) UACC(NONE)
RDEFINE CSF1SKE CLASS(CSFSERV) UACC(NONE)
RDEFINE CSF1SKD CLASS(CSFSERV) UACC(NONE)
RDEFINE CSFOWH  CLASS(CSFSERV) UACC(NONE)
/* permit all, some users, or a group:  */
PERMIT  CSFIQA  CLASS(CSFSERV) ID(*) ACCESS(READ)
...
SETROPTS CLASSACT(CSFSERV)
SETROPTS RACLIST(CSFSERV) REFRESH
```

Note: You must authorize all userids that use ssh including both sshd userids.

ICSF Cipher and MAC Acceleration

- Configuration of `sshd_config` and `ssh_config` Ciphers and MACs options
  - The HOS1130 shipped versions of these files are optimized to choose the best fit with conventional OpenSSH installations along with ICSF acceleration
  - See the guide for information/implications reordering these lists
- Update both z/OS specific configuration files:
  - `/etc/ssh/zos_ssh_config` and `/etc/ssh/zos_sshd_config`

```
# Use either software or ICSF for Ciphers and MACs
CiphersSource any
MACsSource any
```

HCR77A1 performance enhancement option

```
RDEFINE CSF.CSFSERV.AUTH.CSFOWH.DISABLE
    CLASS(XFACILIT) UACC(READ)
RDEFINE CSF.CSFSERV.AUTH.CSFRNG.DISABLE
    CLASS(XFACILIT) UACC(READ)
SETROPTS CLASSACT(XFACILIT)
SETROPTS RACLIST(XFACILIT) REFRESH
```

- Defining these profiles in the XFACILIT class will disable SAF/RACF checks for CSFOWH (hash) and CSFRNG (random number) APIs.
- Since ICSF uses CPACF instructions for these anyway (which can't be protected by SAF/RACF), this is usually an acceptable option.

Verifying ICSF setup

- Run the ssh client under TSO OMVS (new feature!)

```
/SYSTEM/home/user> ssh -vvv myuser@127.0.0.1
…
debug1: zsshVerifyIcsfSetup: ICSF FMID is 'HCR77A0'
debug2: ------------------------------------
debug2: CRYPTO    SIZE      KEY       SOURCE
debug2: ------------------------------------
debug2: AES       256       CLEAR     CPU
debug2: DES       56        CLEAR     CPU
```

Verifying ICSF setup

```
...
debug2: MDC-2      128        NA         CPU
debug2: MDC-4      128        NA         CPU
debug2: MD5        128        NA         SW
debug2: SHA-1      160        NA         CPU
debug2: SHA-2      512        NA         CPU
debug2: TDES       168        CLEAR      CPU
```

Note: `SOURCE=CPU` means CPACF, which is what ICSF uses for SSH Cipher and MAC acceleration.

Note: The strength/size is the largest bit length supported by the facility. In the display above, AES-128, AES-192, and AES-256 are supported via ICSF with CPACF.

Verifying ICSF setup

```
…
debug1: mac_setup_by_alg: hmac-sha1 from source ICSF
debug1: zsshIcsfMacInit (429): CSFPTRC successful:
return code = 0, reason code = 0, handle = 'SYSTOK-
SESSION-ONLY 00000000S   '
```

Note: These messages indicate that ICSF was used for MAC hmac-sha1

Verifying ICSF setup

…

```
debug1: cipher_init: aes128-ctr from source ICSF
debug1: zsshIcsfCipherInit (977): CSFPTRC successful:
return code = 0, reason code = 0, handle = 'SYSTOK-
SESSION-ONLY 00000003S   '
```

Note: These messages indicate that ICSF was used for Cipher aes128-ctr

LE Tuning Recommendations

- Ported Tools OpenSSH uses LE XPLINK runtime libraries (like Java, WebSphere, etc)

See: "Placing Language Environment Modules in LPA .."

  - Add SCEELPA to LPALST
  - Add SCEERUN and SCEERUN2 to LNKLST
  - SCEERUN and SCEERUN2 should be program controlled
  - Implement samples CEE.SCEESAMP(CEEWLPA) and (EDCWLPA) as shipped