

IBM System z9-109 Performance of Cryptographic Operations

(Cryptographic Hardware: CPACF, CEX2C, CEX2A)

© Copyright IBM Corporation 2003. All Rights Reserved

IBM Corporation

Marketing Communications, Server Group

Route 100

Somers, NY 10589

U.S.A.

Produced in the United States of America

All Rights Reserved

IBM, IBM @server, IBM eServer, the IBM logo, the e-business logo, HiperSockets, OS/390, RACF, S/390, z/OS, z/VM, and z9 are trademarks or registered trademarks of International Business Machines Corporation of the United States, other countries or both.

Intel is a registered trademark of the Intel Corporation in the United States, other countries or both.

Linux is a registered trademark of Linux Torvalds.

Other company, product and service names may be trademarks or service marks of others.

IBM may not offer the products, services or features discussed in this document in other all countries in which IBM operates, 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.

Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY, 10504-1785 USA.

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

Performance is in External Throughput Rate (ETR) 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 rates stated here.

Table of Content

IBM System z9-109 Performance of Cryptographic Operations	Page 1
(Cryptographic Hardware: CPACF, CEX2C, CEX2A)	Page 1
Preface	Page 3
1. Introduction	Page 3
2. Cryptographic Hardware supported on z9-109	Page 4
2.1. CP Assist for Cryptographic Function (CPACF)	Page 4
2.2. Crypto Express2 (CEX2) Feature	Page 4
3. Exploitation of Cryptographic Hardware in z9-109	Page 6
3.1. SSL Protocol based Communication	Page 6
4. Performance Information	Page 7
4.1. Definitions	Page 7
4.2. CP Assist for Cryptographic Function (CPACF)	Page 8
4.2.1. CP Assist for Cryptographic Function (CPACF) Performance - Architecture Instruction Interface ('Native')	Page 8
4.2.2. CP Assist for Cryptographic Function (CPACF) Performance - ICSF API Interface	Page 10
4.3. Symmetric Key Advanced Encryption Standard (AES) Performance - ICSF API Interface	Page 12
4.4. Crypto Express2 Performance	Page 14
4.4.1. CEX2 Coprocessor Multiple Data Symmetric Key Performance	Page 15
4.4.2. CEX2 Coprocessor Symmetric Key Performance - Diverse Operations	Page 17
4.4.3. CEX2 Coprocessor PKA Performance	Page 17
4.4.4 CEX2 Accelerator Performance	Page 19
4.5. SSL Protocol Handshake Performance	Page 20
4.5.1. Applicability of SSL Performance Results to a Customer Environment	Page 21
4.5.2. SSL Protocol Performance - System SSL	Page 22
with z/OS V1.7 / Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF)	Page 22
4.5.3. SSL Protocol Performance - Linux Open SSL	Page 23

Preface

The performance information presented in this publication was measured on IBM System z9-109 systems in an unconstrained environment for the specific benchmark with a system control program (operating system) as specified. Many factors may result in variances between the presented information and the information a customer may obtain by trying to reproduce the data. IBM does not guarantee that your results will correspond to the measurement results herein. This information is provided 'as is' without warranty, express or implied.

The performance numbers stated for some of the operations are only for demonstration purposes. When quoting some key length or cryptographic algorithms one may not conclude that IBM implies the key length or cryptographic algorithm are adequate and can therefore be used safely.

The cryptographic functions described here may not be available in all countries and may require special enablement subject to export regulations.

1. Introduction

The purpose of this publication is to provide performance information to the user of cryptographic services on IBM System z9-109 systems. The z9-109 supports the following cryptographic hardware functions:

1. CP Assist for Cryptographic Function (CPACF).
2. Crypto Express2 (CEX2) feature.

The CP Assist for Cryptographic Function delivers cryptographic support on every Central Processor (CP) with Data Encryption Standard (DES), Triple DES (TDES), and Advanced Encryption Standard (AES)-128 bit data encryption/decryption, as well as SHA-1 and SHA-256 hashing.

The Crypto Express2 (CEX2) feature combines the functions of Coprocessor (for secure key encrypted transactions) and Accelerator (for Secure Sockets Layer (SSL) acceleration) modes in a single feature with two PCI-X adapters. New on z9-109, using the HMC console, the PCI-X adapters can be customized as having either two Coprocessors, two Accelerators or one of each. The Crypto Express2 is a replacement for the PCIXCC and PCICA features. All of the equivalent PCIXCC and PCICA functions are implemented in the Crypto Express2 with equivalent or greater performance.

2. Cryptographic Hardware supported on z9-109

2.1. CP Assist for Cryptographic Function (CPACF)

System z9-109 supports the Message Security Assist (MSA) Architecture along with the CP Assist for Cryptographic Function (CPACF). The CP Assist for Cryptographic Function delivers cryptographic support on every Central Processor (CP) with Data Encryption Standard (DES), Triple DES (TDES), and Advanced Encryption Standard (AES) data encryption/decryption, as well as SHA-1 and SHA-256 hashing. As these cryptographic functions are implemented in each CP the potential throughput scales with the number of CPs in the system.

The DES, TDES and AES-128 functions of the CPACF use clear key values. The SHA-1 and SHA-256 hash functions are shipped enabled. The DES, TDES and AES functions require enablement of the CPACF for export control. The CPACF for DES, TDES, AES, SHA-1, and SHA-256 functions can be invoked by problem state instructions defined by an extension of the z9-109 architecture. Support is also available via Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF) in z/OS.

The hardware of the CPACF that performs the symmetric key operations (DES; TDES; AES-128) and SHA functions operates basically synchronous to the CP operations. The CP cannot perform any other instruction execution while a CPACF cryptographic operation is being executed. The CP internal code performs data fetches and stores resultant data while cryptographic operations are executed in the CPACF hardware on a unit basis as defined by the hardware. The hardware has a fixed set up time per request and a fixed operation speed for the unit of operation. Thus maximum throughput can be achieved for larger blocks of data (up to a hardware defined limit).

2.2. Crypto Express2 (CEX2) Feature

The Crypto Express2 (CEX2) feature combines the functions of Coprocessor (for secure key encrypted transactions) and Accelerator (for secure sockets layer SSL acceleration) modes in a single feature with two PCI-X adapters. New on z9-109, using the HMC console, the PCI-X adapters can be customized as having either two Coprocessors, two Accelerators or one of each. The Crypto Express2 is a replacement for the PCIXCC and PCICA features available on zSeries z900 and z990 systems. All of the equivalent PCIXCC and PCICA functions are implemented in the Crypto Express2 with equivalent or greater performance.

There can be a maximum of 8 CEX2 features in a z9-109 system for a total of 16 PCI-X adapters.

When configured in Coprocessor mode, the CEX2 feature supports:

- Secure cryptographic functions
- Use of secure encrypted key values
- Clear key and secure PKA operations
- User defined Extensions (UDX)

The CEX2 in Coprocessor mode provides a security-rich cryptographic subsystem. The tamper-responding hardware is designed to qualify at the highest level under the FIPS 140-2 standard. Specialized hardware performs DES, TDES, RSA, and SHA-1 cryptographic operations in a secure environment. The CEX2 Coprocessor is designed to protect the cryptographic keys and sensitive custom applications. Security relevant cryptographic keys are

encrypted under the Master Key when outside the secure boundary of the CEX2 card. The Master Keys are always kept in battery backed-up memory within the tamper-protected secure boundary of the CEX2 Coprocessor.

The CEX2 Coprocessor also supports the 'clear key' PKA operations that currently are predominantly used to provide SSL protocol communications.

When configured in Accelerator mode, the CEX2 feature provides hardware support to accelerate certain cryptographic operations that occur in the e-business environment. Compute intensive public key operations as used by SSL/TLS protocols can be offloaded from the CP to the CEX2 Accelerator and thus increase system throughput. The CEX2 in Accelerator mode works in 'clear key' mode only.

The operations in the CEX2 are controlled by an on-board microprocessor with memory to hold the controlling program. A security-rich code-loading process enables control program and application program loading into the CEX2. The Linux based control program together with the application program provides for the IBM Common Cryptographic Architecture (CCA) interface for the applications using the CEX2 feature.

The Crypto Express2 executes its cryptographic operations asynchronously to a Central Processor (CP) operation in the z9-109 system. A CP requesting a cryptographic operation from the CEX2 uses the message queuing protocol to communicate with the CEX2. After enqueueing a request to the CEX2, the host operating system will dispense the task that has enqueueing the cryptographic operation and dispatches another task. Thus, processing of the cryptographic operation in the CEX2 will work in parallel to other tasks being executed in a z9-109 CP. A special CP task will poll at fixed time intervals for finished operations of the Crypto Express2, dequeue them, and execute the Release function to cause the redispach of the application waiting for the result of the cryptographic operation. For each PCI-X adapter in the CEX2, up to 8 requests can be waiting in the queue either for execution or waiting with the result of the cryptographic operation to be dequeued by a CP. In the Crypto Express2, several operations can be worked on in parallel.

For z9-109 systems, the Crypto Express2 works with Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF) and the IBM Resource Access Control Facility (RACF®) in a z/OS or OS/390® operating environment to provide cryptographic services with the IBM Common Cryptographic Architecture (CCA) secure key management.

The IBM Common Cryptographic Architecture implementation provides a base on which customer programs can request cryptographic services from the Crypto Express2. For unique customer cryptographic application requirements the Crypto Express2 in Coprocessor mode provides for user-defined extensions (UDX) to the Common Cryptographic Architecture interface.

3. Exploitation of Cryptographic Hardware in z9-109

In the cryptographic application environment it is quite common that an application will not have direct access to the cryptographic hardware. The application requiring a cryptographic service

will call a Programming Interface (API) which is interpreted by some services of the System Control Program.

In the z9-109 using the z/OS System Control Program, most cryptographic hardware can only be used through Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF). ICSF is a standard component of z/OS. It provides cryptographic services in the z/OS environment. ICSF provides the application programming interfaces (APIs) by which applications request cryptographic services. Thus ICSF relieves the application from dealing with the complexity of the cryptographic hardware communication. However, these ICSF services are operating software path lengths which have to be added (from an application's point of view) to the execution time of the cryptographic hardware.

As mentioned in the description of the CPACF cryptographic hardware, an application program can use this hardware by invoking any of the 5 new machine instructions. However, there is also an API call interface available to ICSF. The performance of both modes of operation will be presented in this publication.

3.1. SSL Protocol based Communication

Secure Sockets Layer (SSL) is a communication protocol that was designed to facilitate secure communication over an open communication network, such as the Internet. The SSL protocol is a layered protocol that is intended to be used on top of a reliable transport, e.g. Transmission Control Protocol (TCP/IP). SSL is designed to provide data privacy and integrity by using cryptographic operations and optionally Server and Client authentication based on public key certificates. Once an SSL connection is established between a Client and Server, data communications between Client and Server are transparent to the encryption and integrity added by the SSL protocol. Transport Layer Security (TLS) is the newer version of the SSL protocol.

Executing the SSL/TLS protocols for a Server (or Client) on a z9-109 system will result in a series of cryptographic operations. In the z/OS environment ICSF will either invoke available cryptographic hardware or will execute the cryptographic operation in system software. The SSL/TLS protocol will result in an increase in transaction execution time compared to an unsecure protocol. Some factors contributing to the increase are 1)CP path length (due to the protocol itself and due to operating system support); 2) the symmetric key operation's execution time (either hardware assisted or in software executed on a CP); and 3) the execution time of the public key operations (either hardware assisted (operating in parallel to the CP instruction execution) or in software on a CP). This publication will state the performance in the SSL environment as the maximum number of SSL handshakes the z9-109 can provide as a server within the given system constraints and assess the utilization of the measured system.

The intent for providing capacity information in the SSL environment is to demonstrate the capabilities of a z9-109 system to act as a Web Server providing SSL-compliant communication to a large number of clients. For this purpose the maximum number of SSL connects and data exchanges per second made between the server and all clients are provided for different environments. There is no intention to provide a more detailed performance analysis for this environment.

In this publication, performance/capacity information will be given for running SSL protocol based communication in the following environments:

- z/OS
- Linux

As this performance publication primarily deals with performance of cryptographic operations and Web based communication, the measurements for the SSL environments include only the processing required for the SSL protocol handshake and some data exchange. Explicitly excluded is the processing for the 'business transaction' that in a normal environment would be initiated in the server on behalf of the client's request. As most SSL protocol-based measurements in this report are limited by the processing capacity of the server, in a 'real life' environment the processing for the business transaction would reduce the number of necessary handshakes considerably.

4. Performance Information

4.1. Definitions

The performance information stated in this publication is normally provided on the ICSF API level except when stated otherwise. Measurements were performed with the control program z/OS Version 1 Release 7 (z/OS V1.7) and Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF), except when stated otherwise.

All measurements were performed on an IBM System z9-109. The exact model of the z9-109 system used is stated with each measurement. Most of the measurements were run on a z9-109 Model S18 with 4 Central Processors. If, however, the measurement invokes only one single job the performance behavior is the same as if this measurement were run on a z9-109 Model S18 with only one Central Processor.

For the cryptographic operations that can be used with a variable length of data such as Data Encryption Algorithm (DEA) Standard encryption, the performance is stated for test cases using different data lengths. The length is specified in Bytes ('K' equals 1024, 'M' equals 1,048,576). The resulting data rate is specified in multiples of 1,000,000 Bytes (not 'M').

In order to keep this performance publication at a reasonable length results of measurements are presented using a single cryptographic feature. If multiple cryptographic features are available a statement is made how the performance results scale with usage of multiple features.

4.2. CP Assist for Cryptographic Function (CPACF)

4.2.1. CP Assist for Cryptographic Function (CPACF) Performance - Architecture Instruction Interface ('Native')

All test cases are written in System z9 Assembler Language issuing the System z9 Message Security Assist (MSA) Architecture cryptographic operation instructions as indicated with each group.

The data quoted was from test cases run on a z9-109 Model S18, however, using only one of the CPs. For each cryptographic operation type quoted, there is a statement on scalability of the results if up to 4 CPs are being used. The throughput using N CPs performing the same cryptographic operation is close to N times the throughput of using one CP. The reduction of the measured throughput from N times the throughput of one CP is stated with each measurement.

Terminology Explanation: The term DEA stands for Data Encryption Algorithm which is a block cipher according to the Data Encryption Standard (DES).

DEA Cipher Block Chaining Encipher with Single Length Key (56 Bits)

(System z9 Message Security Assist Architecture instruction: KMC-DEA)

Native: Single DES CBC Encipher (KMC-DEA)			
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec	
64	4813444	308.1	
256	2049426	524.7	
1024	637935	653.2	
4096	170552	698.6	
64K	10629	696.6	
1M	652.1	683.8	

The KMC-DEA operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 1 percent (with 4 Cps).

DEA Cipher Block Chaining Decipher (CBC) with Single Length Key has basically the same performance characteristics as the Encipher operation.

DEA Electronic Code Book Encipher (ECB, without chaining) with Single Length Key has basically the same performance characteristics as the corresponding CBC Encipher operation (increase for small data length up to 17 percent, for larger data length about .4 percent).

DEA Cipher Block Chaining Encipher with Triple Length Key (168 Bits)

(System z9 Message Security Assist Architecture instruction: KMC-TDEA)

Native: Triple DES CBC Encipher (KMC-TDEA)		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	2836286.0	181.5
256	876717.0	224.4
1024	236091.0	243.3
4096	59919.0	245.4
64K	3729.0	244.4
1M	231.3	242.5

The KMC-TDEA operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 1 percent (with 4 Cps).

DEA Cipher Block Chaining Decipher with Triple Length Key has basically the same performance characteristics as the Encipher operation.

AES Cipher Block Chaining Encipher with Single Length Key (128 Bits)

(System z9 Message Security Assist Architecture instruction: KMC-AES)

Native: AES CBC Encipher (KMC-AES)		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	3402855.0	217.8
256	1150398.0	294.5
1024	313562.0	321.1
4096	80149.0	328.3
64K	4986.0	326.8
1M	309.1	324.2

The KMC-AES operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 1 percent (with 4 CPs).

Compute Message Authentication Code with DEA Single Length Key (56 Bits)

(System z9 Message Security Assist Architecture instruction: KMAC-DEA)

Native: MAC with single DES (KMAC-DEA)		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	6850493	438.4
256	2420283.0	619.6
1024	670512.0	686.6
4096	172672.0	707.3
64K	10780.0	706.5
1M	666.0	698.4

The KMAC-DEA operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 1 percent (with 4 CPs).

Compute Message Digest SHA-1

(System z9 Message Security Assist Architecture instruction: KLMD-SHA-1)

Native: SHA-1(KLMD-SHA-1)		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	3066803.0	196.3
256	1571817.0	402.4
1024	534668.0	547.5
4096	145961.0	597.9
64K	9244.0	605.8
1M	573.1	600.9

The KLMD-SHA-1 operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 1 percent (with 4 CPs).

Compute Message Digest SHA-256

(System z9 Message Security Assist Architecture instruction: KLMD-SHA-256)

Native: SHA-256(KLMD-SHA-256)		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	2343608.0	150.0
256	1094264.0	280.1
1024	351152.0	359.6
4096	94960.0	389.0
64K	6024.0	394.8
1M	373.8	392.0

The KLMD-SHA-256 operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 1 percent (with 4 CPs).

4.2.2. CP Assist for Cryptographic Function (CPACF) Performance - ICSF API Interface

All test cases are written in System z9 Assembler Language issuing an API call to ICSF for the cryptographic operation. ICSF will resolve the API call and issue instructions for the cryptographic operation according to the System z9 Message Security Assist (MSA) Architecture as indicated with each group.

The data quoted was from test cases run on a z9-109 Model S18, however, using only one of the CPs. For each cryptographic operation type quoted there is a statement on scalability of the results if multiple CPs are being used. The throughput using N CPs performing the same cryptographic operation is close to N times the throughput of using one CP. The reduction of the measured throughput from N times the throughput of one CP is stated with each measurement.

As the performance measurement results show, all ICSF API interface test cases have lower throughput than the equivalent 'Native' test cases. This is expected because of the additional ICSF path length. As the data length increases, the ICSF path length is a less dominant factor and the throughput is nearly the same as for the 'Native' test cases for large data lengths.

DEA Cipher Block Chaining Encipher with Single Length Key (56 Bits) - ICSF API
(System z9 Message Security Assist Architecture instruction: KMC-DEA)

ICSF API: Single DES CBC Encipher (KMC-DEA) 1 job		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	102131.0	6.54
256	99638.0	25.51
1024	89684.0	91.84
4096	64727.0	126.12
64K	9626.0	630.85
1M	650.1	681.68

The DEA Encipher with Single Length Key operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 11 % for 4 CPs and 64 byte data length and decreases for higher data lengths.

DEA Decipher with Single Length Key has similar performance characteristics as the Encipher operation.

DEA Cipher Block Chaining Encipher with Triple Length Key (168 Bits) - ICSF API
(System z9 Message Security Assist Architecture instruction: KMC-TDEA)

ICSF API: Triple DES CBC Encipher (KMC-TDEA) 1 job		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	100366.0	6.42
256	92989.0	23.81
1024	72099.0	73.83
4096	38012.0	155.70
64K	3601.0	236.00
1M	231.9	243.16

The DEA Encipher with Triple Length Key operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 11 % for 4 CPs and 64 byte data length and decreases for higher data lengths.

DEA Decipher with Triple Length Key has similar performance characteristics as the Encipher operation.

Compute Message Digest SHA-1 - ICSF API

(System z9 Message Security Assist Architecture instruction: KLMD-SHA-1)

ICSF API: SHA-1(KLMD-SHA-1) 1 job		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	79758.0	5.10
256	77741.0	19.90
1024	70930.0	72.63
4096	52423.0	214.72
64K	8307.0	544.40
1M	571.8	599.58

The Compute message Digest SHA-1 operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 12 % for 4 CPs and 64 byte data length and decreases for higher data lengths.

Compute Message Digest SHA-256 - ICSF API

(System z9 Message Security Assist Architecture instruction: KLMD-SHA-2)

ICSF API: SHA-256(KLMD-SHA-256)		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	78651.0	5.0
256	75671.0	19.4
1024	66150.0	67.7
4096	43837.0	179.6
64K	5595.0	366.7
1M	373.2	391.3

The Compute message Digest SHA-2 operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is less than 12 % for 4 CPs and 64 byte data length and decreases for higher data lengths.

4.3. Symmetric Key Advanced Encryption Standard (AES) Performance - ICSF API Interface

The Advanced Encryption Standard (AES) is available as a standard for some time and is now emerging in applications.

With System z9, AES-128 encryption services are provided in the CPACF. IBM continues to provide AES-256 encryption services in the z/OS environment as API calls to ICSF software routines.

The data quoted was from test cases run on a z9-109 Model S18, however, using only one of the CPs. For each cryptographic operation type quoted there is a statement on scalability of the results if multiple CPs are being used. The throughput using N CPs performing the same cryptographic operation is close to N times the throughput of using one CP. The reduction of the measured throughput from N times the throughput of one CP is stated with each measurement.

All measurements were performed with z/OS V1.7 and Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF).

AES128 Encipher (128 bit Key Length) - ICSF API (CPACF)

AES128 Encipher (128 bit key) 1 job		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	100430.0	6.43
256	94890.0	24.29
1024	77976.0	79.85
4096	45302.0	185.56
64K	4782.0	313.39
1M	310.6	325.69

The AES128 Encipher operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is in the order of 11% for 4 CPs and 64 byte data length and decreases for higher data lengths.

AES128 Decipher (128 bit Key Length) - ICSF API (CPACF)

AES128 Decipher (128 bit key) 1 job		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	100146.0	6.41
256	94683.0	24.24
1024	77643.0	79.51
4096	45206.0	185.16
64K	4755.0	311.62
1M	310.5	325.58

The AES128 Decipher operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is in the order of 11% for 4 CPs and 64 byte data length and decreases for higher data lengths.

AES256 Encipher (256 bit Key Length) - ICSF API (Software)

AES256 Encipher (256 bit key) in software		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	85376.0	5.46
256	61013.0	15.62
1024	28341.0	29.02
4096	8998.0	36.86
64K	613.2	40.19
1M	38.4	40.21

The AES256 Encipher operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is in the order of 9% for 4 CPs and 64 byte data length and decreases for higher data lengths.

AES256 Decipher (256 bit Key Length) - ICSF API (Software)

AES256 Decipher (256 bit key) in software		
Data Length (Bytes)	Operations/sec	x10**6 Bytes/sec
64	75975.0	4.86
256	56292.0	14.41
1024	26408.0	28.07
4096	8964.0	36.72
64K	618.5	40.53
1M	38.8	40.63

The AES256 Decipher operation scales with the number of CPs executing multiple jobs with the same operation. The reduction is in the order of 7% for 4 CPs and 64 byte data length and decreases for higher data lengths.

4.4. Crypto Express2 Performance

The Crypto Express2 feature is designed to satisfy high-end server security requirements. The Crypto Express2 feature, with two PCI-X adapters, is configurable and can be defined for secure key encrypted transactions (Coprocessor – the default) or SSL acceleration (Accelerator). Crypto Express2 executes the functions that were previously offered by the PCICA and PCIXCC features, performing hardware acceleration for SSL transactions and clear key RSA operations. Like its predecessors, the Crypto Express2 feature has been designed to satisfy the security requirements of an enterprise server. The PCIXCC, PCICC, and PCICA features are not supported on z9-109.

When configured as a Coprocessor, the PCI-X adapter is designed to provide security-rich cryptographic operations to be used by System z9 host application programs. The Coprocessor mode offers security for symmetric key and public key operations. In this case the cryptographic keys are encrypted under the corresponding Master Keys when outside the secure boundary of the PCI-X adapter.

When configured as an Accelerator, the PCI-X adapter is designed to provide high speed acceleration of RSA operations in ‘clear key’ mode, providing security rich communication for Web site-based applications which utilize the SSL or TLS protocol. It is current practice to execute the public key operation, incurred during set up of an SSL session, in ‘clear key’ mode.

The connection of the CEX2 feature via the PCIX bus to the z9-109 Central Processors (CPs) incurs latency and data transmission time. Because of this connection to the z9-109 CPs, the CEX2 operates asynchronous to the z9-109 CPs.

There can be a maximum of 8 CEX2 features in a z9-109 system, each CEX2 feature containing two PCI-X adapters.

4.4.1. CEX2 Coprocessor Multiple Data Symmetric Key Performance

This chapter deals with CEX2 Coprocessor cryptographic operations with a user supplied length of data as, e.g., DES operations.

All test cases are written in System z9 Assembler Language issuing an API call to ICSF for the cryptographic operation. ICSF will resolve the API call and handle the communication with the CEX2 Coprocessor feature which does the actual cryptographic processing. The symmetric key that is used for the cryptographic operation is encrypted under the corresponding Master Key which in turn is kept in the secure boundary of the PCI-X adapter.

The throughput for the cryptographic operations using the CEX2 Coprocessor for multiple data symmetric key operations is considerably less than the throughput for the corresponding functions using the CP Assist for Cryptographic Function (CPACF) hardware. For this type of cryptographic operations the CEX2 Coprocessor feature should be used only when the security requirements for the application require it. Be aware that in the tables of this chapter the rates are quoted in thousands of Bytes, not in millions of bytes as in previous tables.

The data quoted was from test cases run on a z9-109 Model S18 using 1 job that performs the cryptographic operation. For each cryptographic operation type quoted there is a statement on scalability of the results if multiple jobs are being used. The increase of measured throughput using 7 jobs is exemplified for the Single DES CBC Encipher operation.

The performance numbers are from measurements with z/OS V1.7 including Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF).

CEX2 Coprocessor DEA Cipher Block Chaining Encipher with Single Length Key (56 Bits)

CEX2C (one job): Single DES CBC Encipher		
Data Length (Bytes)	Operations/sec	x10**3 Bytes/sec
64	914.90	58.55
256	914.70	234.2
1024	906.70	928.5
4096	616.30	2524.4
64K	63.43	4156.9
1M	4.11	4307.6

The above table provides measurement results for an environment where one job was continuously executing the cryptographic operation using one CEX2 Coprocessor card. As mentioned, the execution of the cryptographic operation in the CEX2C card is asynchronous to the z9-109 Central Processor (CP) execution. As only one job is run on the CP the next cryptographic operation is started only when the result of the previous cryptographic operation has been received by the CP. Thus there is a considerable delay before the next cryptographic operation can be initiated by the host CP. This inefficiency is removed when the host program consists of several jobs requesting the same cryptographic operation repetitively. The CEX2C adapter's multitasking capability allows for enqueueing and dequeuing of requests in parallel with cryptographic operations being performed. A measurement environment using several parallel jobs highlights better the throughput capacity of the CEX2C adapter whereas the 'single job' measurement environment is better suited to highlight the delay an application experiences waiting for the result of the cryptographic operation performed in the CEX2C.

CEX2C (seven jobs): Single DES CBC Encipher		
Data Length (Bytes)	Operations/sec	x10**3 Bytes/sec
64	1393.0	89.15
256	1304.0	333.8
1024	1042.0	1067.0
4096	827.8	3390.7
64K	78.34	5134.1
1M	5.06	5307.9

The throughput with N CEX2C adapters with a sufficient number of jobs repetitively requesting the same cryptographic operation for Single DES, Triple DES, and Single DES Message Authentication (MAC) (see the following tables) is close to N times the throughput of one CEX2C adapter with 7 jobs (as exemplified above).

CEX2 Coprocessor DEA Cipher Block Chaining Encipher with Triple Length Key (168 Bits)

CEX2C (one job): Triple DES CBC Encipher		
Data Length (Bytes)	Operations/sec	x10**3 Bytes/sec
64	914.8	58.55
256	914.7	234.2
1024	663.3	679.5
4096	615.7	2521.9
64K	55.91	3664.1
1M	3.60	3769.6

The throughput for seven jobs for CEX2C TDES is in the order of 1.2 times to 1.6 times higher than for one job.

CEX2 Coprocessor Message Authentication Code with DEA Single Length Key (56 Bits)

CEX2C (one job): MAC with single DES		
Data Length (Bytes)	Operations/sec	x10**3 Bytes/sec
64	915.4	58.59
256	915.4	234.3
1024	914.6	936.6
4096	911.6	3733.9
64K	86.65	5678.7
1M	5.87	6155.1

The throughput for seven jobs for CEX2C MAC is in the order of 1.2 to 1.5 times higher than for one job, the lower number applying to large data length and the higher to small data lengths.

4.4.2. CEX2 Coprocessor Symmetric Key Performance - Diverse Operations

The following table gives the performance in maximum number of operations per second for one CEX2 Coprocessor for some selected symmetric key operations.

CEX2C Symmetric Key Operations - Examples	Ops/s	Ops/s
	1 job	7 jobs
Key Generate (operational DES KEYGENKY key)	620	935
Clear PIN Generate Alternate (DES OPINENC + DES PINGEN keys)	815	990
Clear PIN Generate (16 digits) (DES PINGEN key)	917	1,393

The throughput with N CEX2C adapters with a sufficient number of jobs repetitively requesting the same cryptographic operation for the examples in the table above is close to N times the throughput of one CEX2C adapter with 7 jobs.

4.4.3. CEX2 Coprocessor PKA Performance

The CEX2 Coprocessor is designed to offer good Public Key Algorithm (PKA) cryptographic operation performance in addition to the high-security environment. The PKA performance is listed for RSA key modulus length of 512 bits, 1024 bits (1K bits), and 2048 bits (2K bits).

The numbers quoted for performing the Public Key Decrypt (PKD) cryptographic operation (using the Private Exponent) are either through the Chinese Remainder Theorem (CRT) Format or the Modulus Exponent (ME) Format. The PKD operation uses the private key in 'clear key' mode.

For the Public Key Encrypt (PKE) cryptographic operation ICSF always uses an RSA public key with the Modulus Exponent (ME) Format. The modulus is according to the length specified and the (Public) Exponent has the value of 65537 which in hexadecimal notation is X'10001' (with leading zeros up to the length of the modulus).

For the Digital Signature Generate (DSG) and the Symmetric Key Import (SYI) cryptographic operations the PKA keys (signature key or encryption key) are encrypted under the corresponding master key.

The performance numbers are from measurements with z/OS V1.7 including Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF) invoking the operation via the ICSF API according to the PKCS-1.2. Standard. Measurements were performed on a z9-109 Model S18.

CEX2 Coprocessor PKA Performance

CEX2C on z/OS V1.7 (ICSF level: WD#5)				
Public Key Decrypt (PKD), Public Key Encrypt (PKE) Digital Signature Generate (DSG), Digital Sign. Verify (DSV) Symmetric Key Import (encrypted with RSA key) (SYI)				
	2094-104	2094-104	2094-104	2094-104
CEX2C	1	1	2	4
Jobs	1	7	14	28
	Operations/sec	Operations/sec	Operations/sec	Operations/sec
PKD--CRT, 512 bit	890	1110	2216	4428
PKD--CRT, 1024 bit	615	1004	2006	4004
PKD--CRT, 2048 bit	269	466	932	1861
PKD--ME, 512 bit	615	1087	2184	4348
PKD--ME, 1024 bit	466	926	1850	3693
PKE, 512 bit	909	1221	2440	4871
PKE, 1024 bit	870	1012	2018	4018
PKE, 2048 bit	616	772	1547	3094
DSG--CRT, 512 bit	896	1123	2252	4371
DSG--CRT, 1024 bit	615	1005	2026	4042
DSG--CRT, 2048 bit	269	466	932	1862
DSV--ME, 512 bit	913	1357	2715	5402
DSV--ME, 1024 bit	913	1268	2542	5069
SYI--CRT, 512 bit	616	844	1689	3375
SYI--CRT, 1024 bit	558	797	1595	3184

The PKA cryptographic operation throughput with N CEX2C adapters with a sufficient number of jobs repetitively requesting the same cryptographic operation for the examples in the table above is close to N times the throughput of one CEX2C adapter with 7 jobs (as stated above) except for DSG-CRT with 512 bit length which gave the factor of 3.8 for four CEX2C adapters.

PKA RSA Key Generate

The CEX2 Coprocessor also offers services to generate PKA RSA Keys. The PKA RSA Key Generate performance is listed for RSA key modulus length of 512 bits, 1024 bits (1K bits), and 2048 bits (2K bits) dependent on the Format, either the Chinese Remainder Theorem (CRT) Format or the Modulus Exponent (ME) Format.

PKA Key Generation is a compute intensive operation. The table below specifies the number of Key generations per second provided by one CEX2 Coprocessor.

CEX2 Coprocessor PKA RSA Key Generation Performance

CEX2C PKA RSA Key Generate	
	Operations/sec
External CRT, 512bit	3.42
External CRT, 1024bit	1.71
External CRT, 2048bit	0.66
Internal ME, 512bit	4.06
Internal ME, 1024bit	1.91

4.4.4 CEX2 Accelerator Performance

The CEX2 Accelerator configuration mode is designed to offer fast Public Key Algorithm cryptographic (PKA) operations. The PKA performance is listed for RSA key modulus length of 512 bits, 1024 bits (1K bits), and 2048 bits (2K bits). The performance numbers are from measurements with z/OS V1.7 including Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF) invoking the operation via the ICSF API according to the PKCS-1.2 Standard.

Quoted are the numbers performing the Public Key Decrypt (PKD) cryptographic operation which uses the Private Exponent either through the Chinese Remainder Theorem (CRT) Format or the Modulus Exponent (ME) Format.

For the Public Key Encrypt (PKE) cryptographic operation ICSF always uses an RSA public key with the Modulus Exponent (ME) Format. The modulus is according to the length specified and the (Public) Exponent has the value of 65537 which in hexadecimal notation is X'10001' (with leading zeros up to the length of the modulus)

CEX2 Accelerator PKA Performance

CEX2A Public Key Decrypt (PKD) and Public Key Encrypt (PKE) (z/OS V1.7 ,ICSF: WD#5)			
2094 CPs	4	4	4
CEX2A Adapters	1	1	4
Jobs	1	8	32
	Operations/sec	Operations/sec	Operations/sec
PKD--CRT, 512 bit	1744	10199	38531
PKD--CRT, 1024 bit	1741	3334	13297
PKD--CRT, 2048 bit	376	456	1821
PKD--ME, 512 bit	1745	3371	13451
PKD--ME, 1024 bit	619	920	3673
PKE, 512 bit	895	6790	24915
PKE, 1024 bit	895	6797	23937
PKE, 2048 bit	895	6788	24468

The first result column of the above table is for measurements where one job was continuously executing the cryptographic operation using one CEX2 Accelerator card. As mentioned, the execution of the cryptographic operation in the CEX2 Accelerator is asynchronous to the z9-109 Central Processor (CP) execution. As only one job is run on the CP the next cryptographic operation is started only when the result of the previous cryptographic operation has been received by the CP. The single job measurement indicates the delay an application would experience waiting for the result of the cryptographic operation.

The second result column of the above table is for measurements where eight jobs were continuously executing the same cryptographic operation using one CEX2 Accelerator card. The increased throughput is due to the fact that tasks are always available for execution in the CEX2 Accelerator card due to the parallel threads that run in the z9-109 CPs. Thus the full capability of the CEX2 Accelerator card for parallel execution of the cryptographic operation can be utilized.

The third column of the above table is for measurements where 32 jobs were continuously executing the same cryptographic operation using 4 CEX2 Accelerator cards. The results show the scalability of the throughput when multiple CEX2A adapters are used in one z9-109 system.

4.5. SSL Protocol Handshake Performance

The SSL handshake protocol is used to negotiate the secure attributes of a session between Client and Server. This process establishes Protocol Version, Session Identification (SID), Authentication (authentication of the Client is optional), and a symmetric key to help protect the data transmitted between Server and Client. The attributes of an established session can be kept as Session Identification in a Client and/or Server cache for later reuse. This may be of interest as establishing a session is a compute intensive process and requires on the Server side a PKA Private Key operation. This Public Key Decrypt (PKD) on the Server can be performed either in software or may be assisted by cryptographic hardware. In the presented measurements on the z9-109 the PKD operation will be routed for execution to the CEX2 Coprocessor or CEX2 Accelerator adapter, if available in the configuration. For all presented measurements the PKD operation is in 'clear key' mode which is currently the predominate usage for SSL protocol communications.

For all SSL protocol performance measurements in this publication the following applies:

- Measurements were performed on a z9-109 system with 4 CPs as a Server.
- The performance data is for the server only. The server was driven to a maximum utilization by increasing the number of client systems (on separate systems) until some system resource came to its limits.
- The key length for the Public Key operation is 1024 bits. The SSL data encryption is Triple DES (168 bits) and SHA cipher except when stated otherwise. This SSL data symmetric key encryption for TDES and SHA is executed in CPACF hardware.
- One packet of 2048 Bytes is used as Send Bytes and Receive Bytes.
- The SSL protocol handshake is the pure handshake with the transfer of one 2048 Bytes data packet.

Legend for all SSL Performance Tables:

Caching Session ID: If the SID is cached the initial handshake process is avoided. If the SID is not cached the initial handshake has to be performed for every new connection between Client and Server.

Handshake: If the Session ID is 100 % cached the initial handshake is always avoided. If the handshake has to be performed the compute intensive PKD operation, then necessary on the server, can be performed in System SSL software or with hardware on a CEX2 Accelerator or CEX2 Coprocessor adapter.

Client Authentication: The authentication of the Client is optional in the SSL protocol.

External Throughput Rate (ETR): Number of handshakes performed per second.

CPU Utilization %: Average utilization of the z9-109 system Central Processors

Crypto Utilization %: Average utilization of the CEX2 Accelerator or CEX2 Coprocessor adapters.

4.5.1. Applicability of SSL Performance Results to a Customer Environment

As mentioned, the measurements for the SSL protocol handshake include the 'pure' handshake and the transfer of one 2048 Bytes encrypted data packet. There is no instruction processing for the application which means there is no instruction processing that results from a 'business transaction' with e.g. a query and potential update of a data base. The performance numbers provided give guidelines only on the additional system resources required if an existing On-line transaction environment were converted by replacing the 'unchecked' transaction protocol by an SSL protocol for the communication between Client and Server.

The performance measurement results clearly suggest using cryptographic hardware for improved throughput in the transaction rate if more than a few transactions per second are expected to be handled using an SSL protocol transaction. Furthermore, the measurement results show the throughput with one CEX2 Accelerator adapter being in the order of three times the throughput as with one CEX2 Coprocessor adapter in the SSL environment. Thus for high SSL protocol transaction rate environments, CEX2 Accelerator is the preferred configuration mode for a z9-109 system.

The resource consumption in system processing power for one SSL protocol handshake is in the order of 1/6000 of the system (see table below) in the z/OS environment for a z9-109 Model S18 with 4 Central Processors and 2 CEX2 features (4 CEX2 Accelerator cards).

If the transaction were to be 'secured' by an SSL protocol and the server portion were run on a z9-109 system the maximum transaction rate achieved on that server without the SSL protocol would be reduced by the portion of processing capacity that is required for the Server SSL protocol path length.

4.5.2. SSL Protocol Performance - System SSL with z/OS V1.7 / Cryptographic Support for z/OS V1R6/R7 and z/OS.e V1R6/R7 (ICSF)

z9-109 Model S18 (4 Central Processors)

Caching SID	Handshake	Client Auth.	ETR	CPU Util. %	Crypto Util. %
100%	Avoided	no	8,477	97.8	NA
no	Software	no	419	100	NA
no	8 CEX2C	no	6,228	99.4	79.4
no	4 CEX2A	no	6,201	99.9	49.5
no	4 CEX2A	yes	3,933	90.4	47.2

Using the CEX2C cryptographic hardware compared to using System SSL Software (second and third line in the above table) produces an increase in throughput (number of SSL protocol handshakes) of 14.8 times.

The 6,228 ETR (third row) represents the maximum number of SSL handshakes that can be supported with this system because the 4 Central Processors are nearly 100% utilized. The average utilization of the 8 CEX2C adapters is 79.4%, indicating that the 8 CEX2C adapters could process more than 7800 SSL handshakes before reaching 100% utilization.

The fourth row shows that a similar throughput rate can be achieved with CEX2A adapters. With the CEX2A configuration mode, only 4 adapters were used and the average utilization of the CEX2A adapters was 49.5%, indicating that the 4 CEX2A adapters could process more than 12,500 SSL handshakes before reaching 100% utilization.

If Client authentication is required the throughput of the server is considerably reduced, as shown in row 5 of the above table.

4.5.3. SSL Protocol Performance - Linux Open SSL

For all Linux Open SSL measurements the following applies:

- Linux System Level: SLES9 SP3
- Linux Kernel Level: 2.6.5
- Open SSL Code Level: 0.9.7d
- z9-109 Crypt Level: 1.3.3

- No Client Authentication

Linux Open SSL - Native Measurements

Caching SID	Handshake	ETR	CPU Utilization %
no	Software	279	97.8
no	8 CEX2C Cards	8,818	95.6
no	4 CEX2A Cards	8,704	98.7

Using the CEX2 Coprocessor hardware provides an increase in throughput (number of SSL protocol handshakes) of 31.6 times the throughput of using Open SSL Software (first and second line in the above table).

Using the CEX2 Accelerator hardware provides a similar ETR as with CEX2 Coprocessor hardware, but fewer adapters are required to support the transaction rate.