GUIDE SHARE EUROPE
DEUTSCHE REGION

V13
Kryptographie mit Linux for System z - Erfahrungen und Ausblick

Dortmund, 28.April 2009

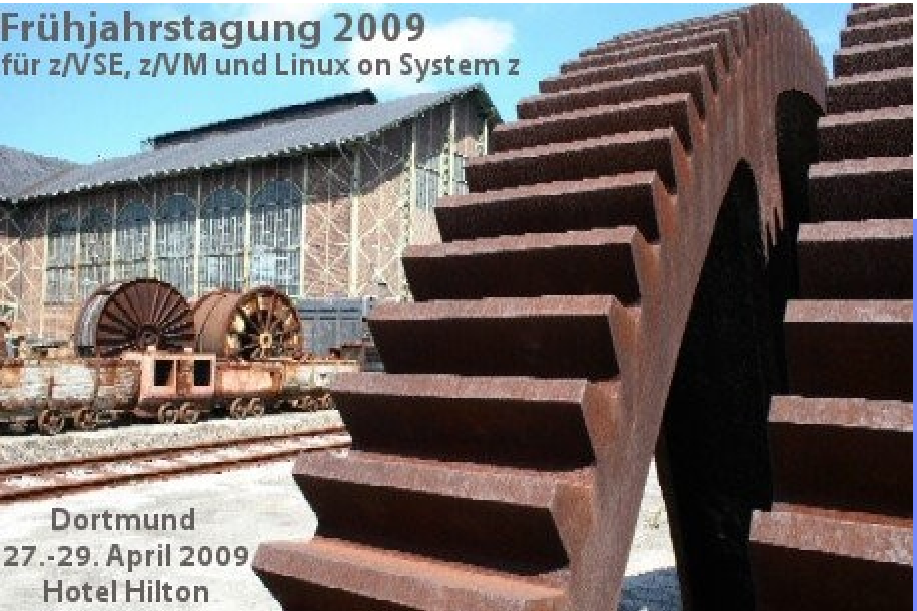
Dr. Manfred Gnirss

Technical Marketing Competence Center Europe

R&D Support Support Centers Boeblingen

IBM Deutschland Research & Development GmbH

gnirss@de.ibm.com



Trademarks

The following are trademarks of the International Business Machines Corporation in the United States, other countries, or both.

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

*, AS/400®, e business(logo)®, DBE, ESCO, eServer, FICON, IBM®, IBM (logo)®, iSeries®, MVS, OS/390®, pSeries®, RS/6000®, S/30, VM/ESA®, VSE/ESA, WebSphere®, xSeries®, z/OS®, zSeries®, z/VM®, System i, System i5, System p, System p5, System x, System z, System z9®, BladeCenter®

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.
IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

* 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

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- Setup for cryptographic hardware support on z10
- Access of cryptographic hardware support with Linux for System z
- In-kernel cryptographical support
- Linux applications using HW cryptographic Support
- New tool
- Cryptographic support with Java on Linux for System z
- NSS Network Security Services
- openssh
- Secure key cryptography

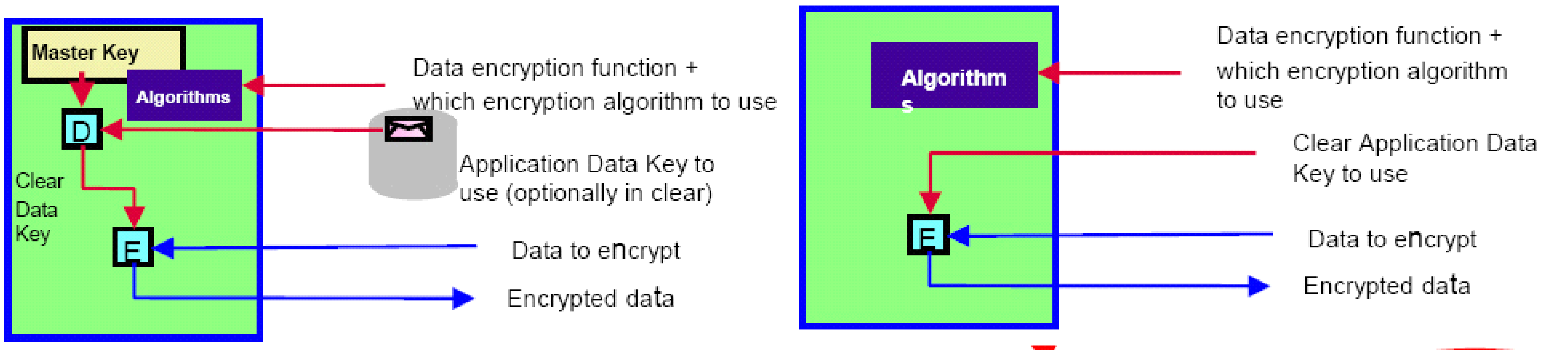
Clear Key and Secure Key Support

secure coprocessor

tamperproof hardware
(CCF, PCICC or PCIXCC/Crypto Express2)

non-secure coprocessor or 'accelerator'

PCICA, CPACF



Master Key

Algorithms

D

Clear Data Key

E

Data encryption function + which encryption algorithm to use

Application Data Key to use (optionally in clear)

Data to encrypt

Encrypted data

Algorithms

E

Data encryption function + which encryption algorithm to use

Clear Application Data Key to use

Data to encrypt

Encrypted data

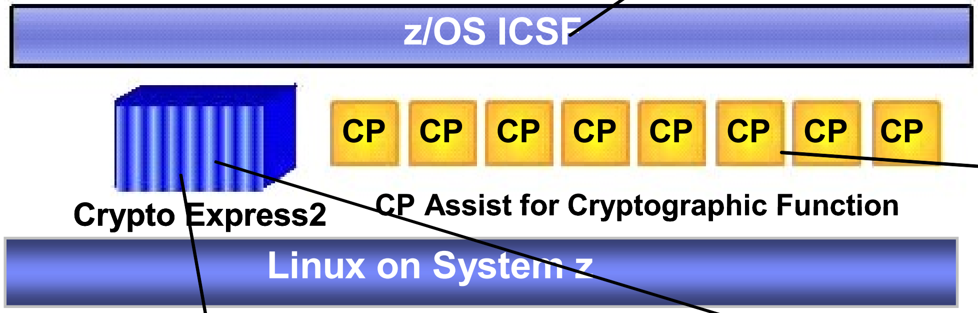
CCF, PCICC evaluated FIPS 140-1 level 4
PCIXCC/Crypto Express2
FIPS 140-2 level 4 certification in process

Focus here is to provide as much throughput as possible

Very sophisticated physical design, requires additional logic

PCIXCC has a two Master Keys: one to protect symmetric keys and another one to protect asymmetric keys

System z Cryptography Features



Methodology to help protect and manage keys

  - Highly secure and available key data store
  - Long term key management
  - Disaster recovery capabilities
- Over 15 years of production use

Encryption acceleration

- Included in every System z general purpose engine
- Very high performance TDES, AES -128 (z9), AES-256 (z10) and SHA-256

SSL acceleration

- Offloads compute-intensive RSA public & private-key cryptographic operations

For secure key processing

- “Tamper-resistant” packaging
- Important for highly secure encryption processing
  - ATM and POS support
  - Securing public and private keys
  - CVV validation, Trusted Key Entry, TDES
- **Lower entry with single port card on System z BC**
- **Linux on System z support**
- Holds Industry’s top hardware rating - FIPS 140-2 Level 4

new

Agenda

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- **Setup for cryptographic hardware support on z10**
- Access of cryptographic hardware support with Linux for System z
- In-kernel cryptographical support
- Linux applications using HW cryptographic Support
- New tool
- Cryptographic support with Java on Linux for System z
- NSS Network Security Services
- openssh
- Secure key cryptography

System z Cryptographic Setup

Careful planning

•Esp. if you do not want too often perform LPAR Activate and Deactivate

•Which adapter / domain to which LPAR

•Which LPAR for cyrpto configuration via TKE

•(Master-) key

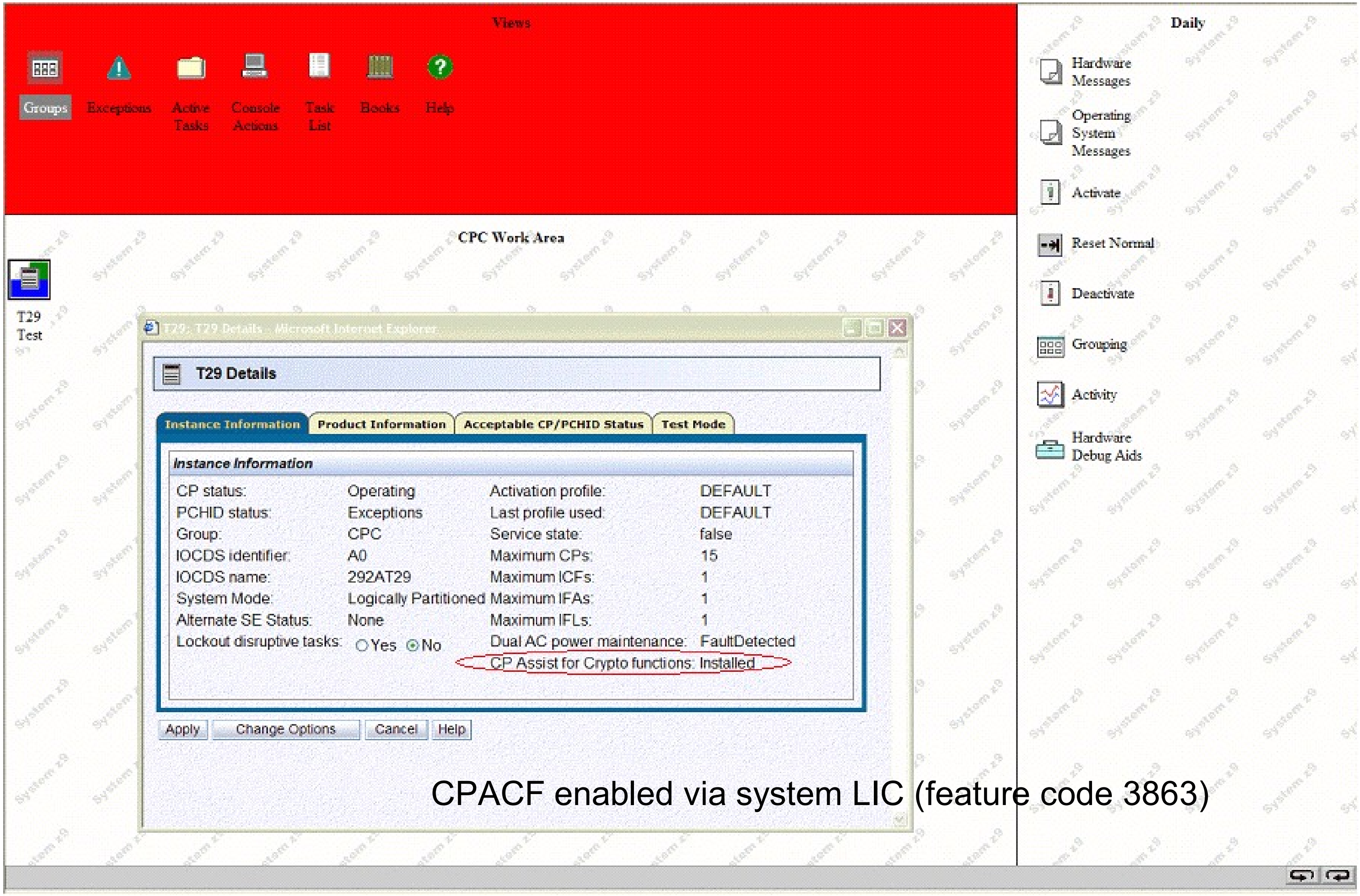
•Up to 8 features with 2 PCI-X adapters (cards, processors)
(1 PCI-X adapter per Crypto Express2-1P)

•How many coprocessors, how many accelerators,

•Sharing, redundancy

•You need LIC internal feature 3863 (Crypto Enablement feature)
  •By default: System z is delivered without this feature!
  •Installation is non-disruptive.

Crypto enablement feature is installed



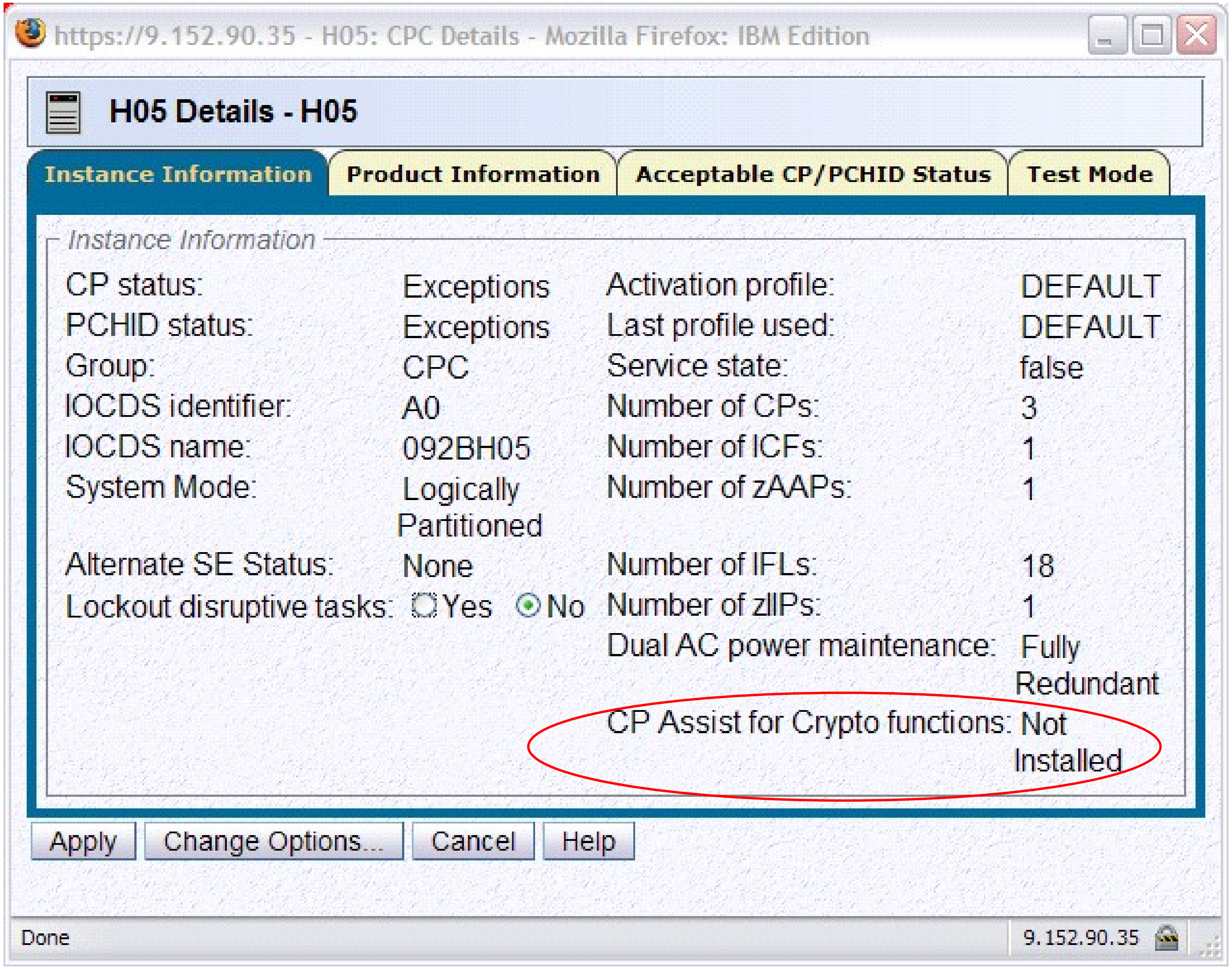
T29 Details

Instance Information

CP status:	Operating	Activation profile:	DEFAULT
PCHID status:	Exceptions	Last profile used:	DEFAULT
Group:	CPC	Service state:	false
IOCDS identifier:	A0	Maximum CPs:	15
IOCDS name:	292AT29	Maximum ICFs:	1
System Mode:	Logically Partitioned	Maximum IFAs:	1
Alternate SE Status:	None	Maximum IFLs:	1
Lockout disruptive tasks:	○ Yes ● No	Dual AC power maintenance:	FaultDetected
		CP Assist for Crypto functions:	Installed

CPACF enabled via system LIC (feature code 3863)

Example: Crypto enablement feature is not installed



https://9.152.90.35 - H05: CPC Details - Mozilla Firefox: IBM Edition

H05 Details - H05

Instance Information | Product Information | Acceptable CP/PCHID Status | Test Mode

Instance Information

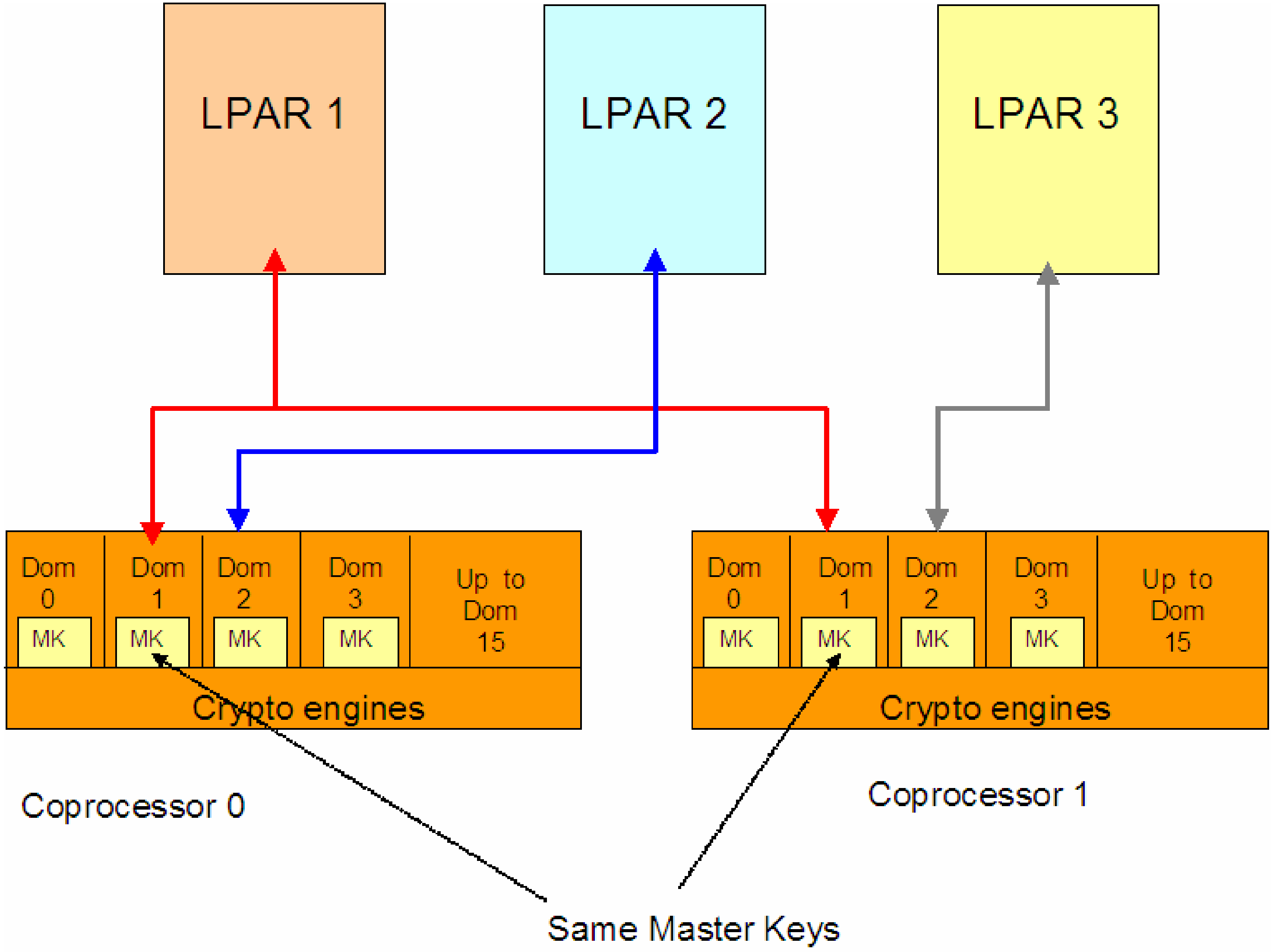
CP status:	Exceptions	Activation profile:	DEFAULT
PCHID status:	Exceptions	Last profile used:	DEFAULT
Group:	CPC	Service state:	false
IOCDS identifier:	A0	Number of CPs:	3
IOCDS name:	092BH05	Number of ICFs:	1
System Mode:	Logically Partitioned	Number of zAAPs:	1
Alternate SE Status:	None	Number of IFLs:	18
Lockout disruptive tasks:	○ Yes ● No	Number of zIIPs:	1
		Dual AC power maintenance:	Fully Redundant
		CP Assist for Crypto functions:	Not Installed

Crypto Express2: Coprocessor or accelerator

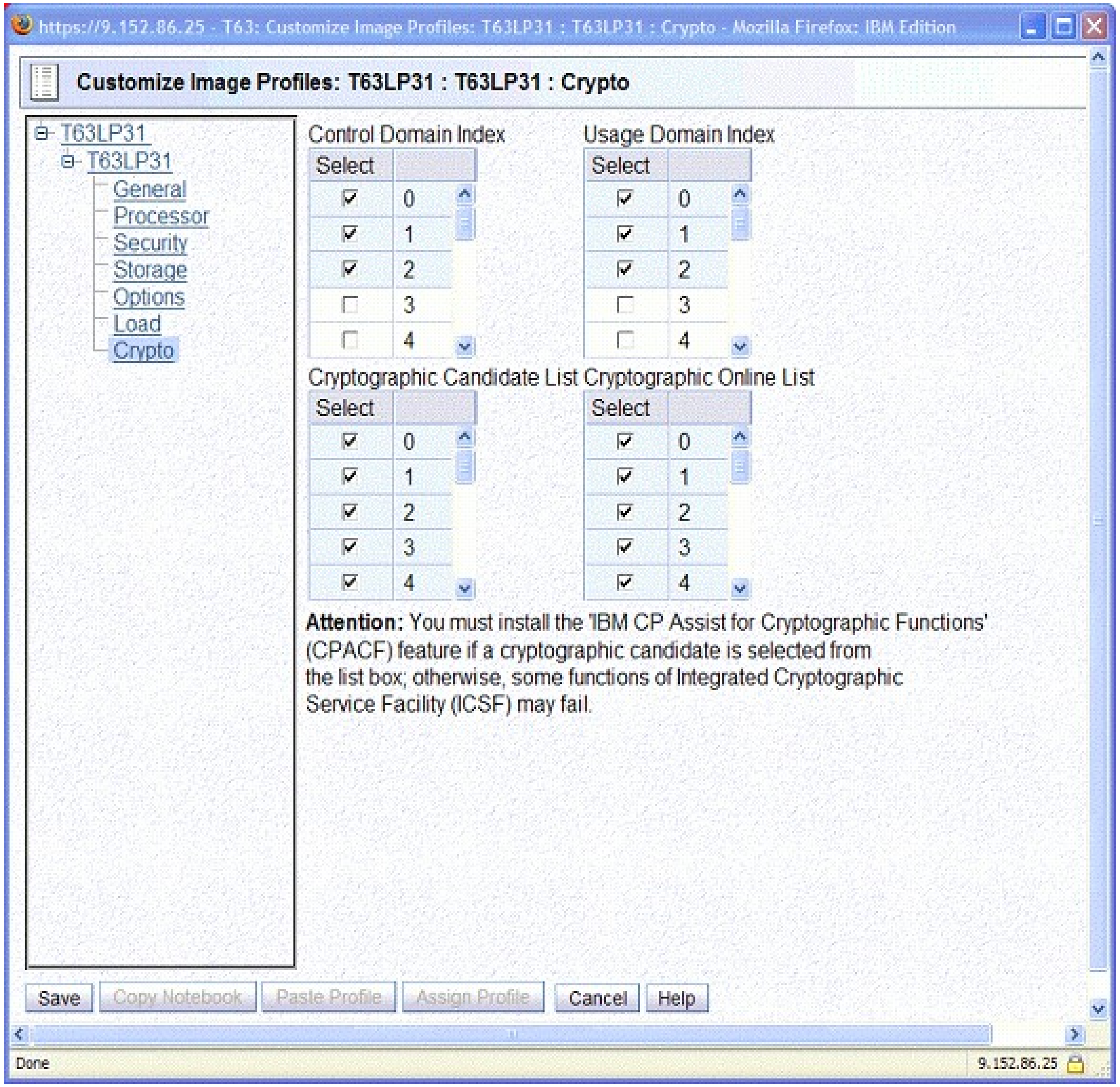
	Adapter Type	Domain Index 0	Domain Index 1	Domain Index 2	…/…	Domain Index 14	Domain Index 15
PCI-X Adapter 0	CEX2C/A	LP00 LP02	LP05	LP04		LP04	
PCI-X Adapter 1	CEX2C/A	LP01 LP02					
PCI-X Adapter 2	CEX2C/A	LP00					
…/…							
PCI-X Adapter 14	CEX2C/A						
PCI-X Adapter 15	CEX2C/A						

- **LP04 and LP05 use different domain numbers for Adapter 0: no conflict (adapter-number.domain-number is unique accros partitions).**
- **LP00 and LP01 use domain 0, but different adapters: no conflict, can be concurrently active.**
- **LP02 uses domain 0 on a set of adapters already defined to LP00 and LP01: LP02 can ot be concurrently active with LP00 or LP01. May be a valid backup configuration.**

Assign Crypto Domain to LPARs



Customize Image Profile



https://9.152.86.25 - T63: Customize Image Profiles: T63LP31 : T63LP31 : Crypto - Mozilla Firefox: IBM Edition

Customize Image Profiles: T63LP31 : T63LP31 : Crypto

- T63LP31
  - T63LP31
    - General
    - Processor
    - Security
    - Storage
    - Options
    - Load
    - Crypto

Control Domain Index

Select	
☑	0
☑	1
☑	2
□	3
□	4

Usage Domain Index

Select	
☑	0
☑	1
☑	2
□	3
□	4

Cryptographic Candidate List

Select	
☑	0
☑	1
☑	2
☑	3
☑	4

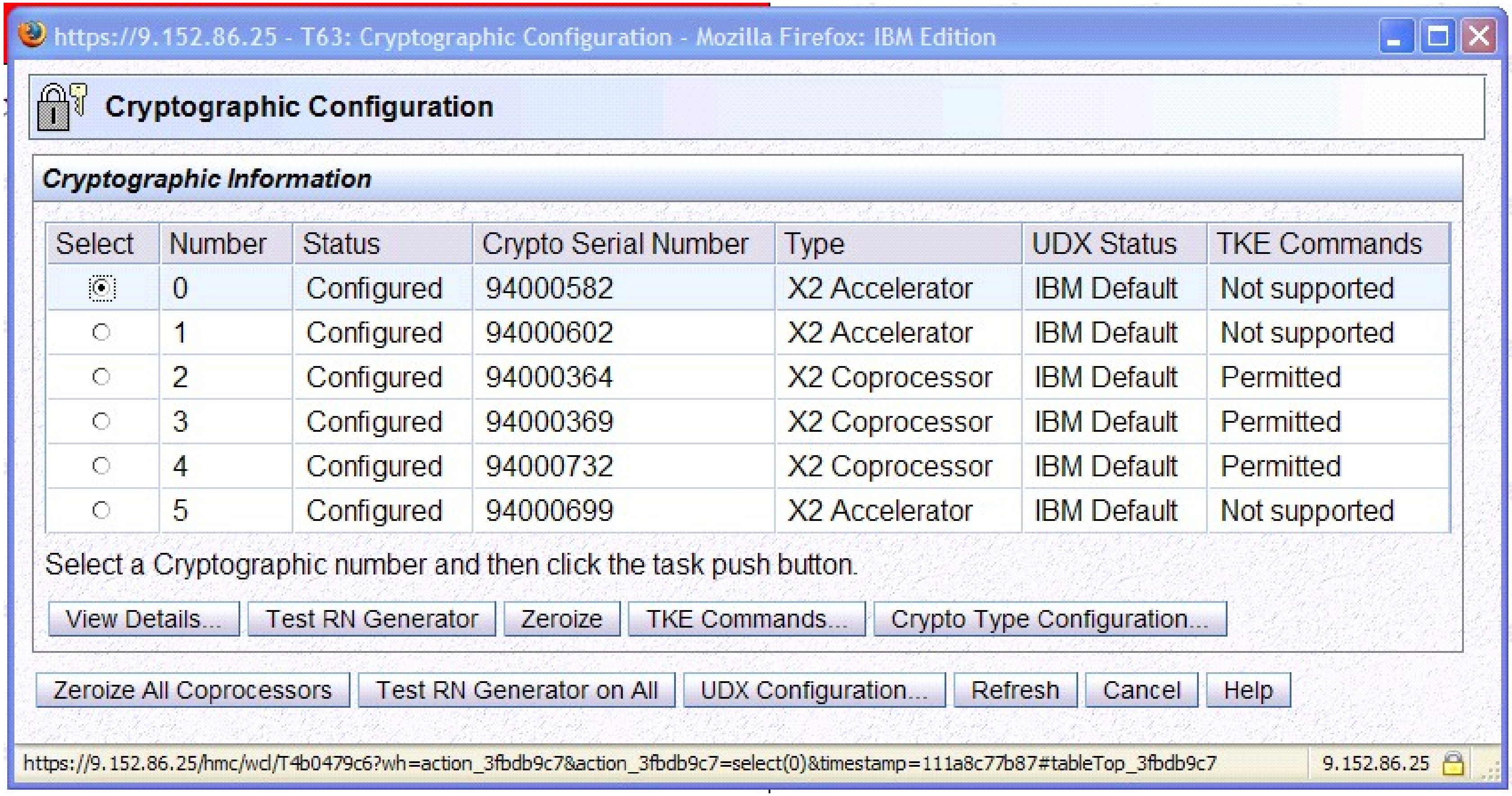
Cryptographic Online List

Select	
☑	0
☑	1
☑	2
☑	3
☑	4

Attention: You must install the 'IBM CP Assist for Cryptographic Functions' (CPACF) feature if a cryptographic candidate is selected from the list box; otherwise, some functions of Integrated Cryptographic Service Facility (ICSF) may fail.

- Combination of Usage Domain Index and PCI-X adapter number must be unique across all active partitions! (exception for backup configurations).
- To newly installed crypto coprocessors numbers are assigned sequentially (during power-on-reset).
- For non-disruptive concurrent installation of a Crypto Express2 feature, out-of-sequence number (from unused range) can be assigned (please inform IBM installation team).
- To dynamically enable a PCI-X adapter to a partition, you need
  - at least 1 usage domain index
  - and coprocessor number must be in the candidate list.
- Changes need partition deactivate-activate! (z9)

Crypto Express2: Coprocessor or accelerator


https://9.152.86.25 - T63: Cryptographic Configuration - Mozilla Firefox: IBM Edition

Cryptographic Configuration

Cryptographic Information

Select	Number	Status	Crypto Serial Number	Type	UDX Status	TKE Commands
●	0	Configured	94000582	X2 Accelerator	IBM Default	Not supported
○	1	Configured	94000602	X2 Accelerator	IBM Default	Not supported
○	2	Configured	94000364	X2 Coprocessor	IBM Default	Permitted
○	3	Configured	94000369	X2 Coprocessor	IBM Default	Permitted
○	4	Configured	94000732	X2 Coprocessor	IBM Default	Permitted
○	5	Configured	94000699	X2 Accelerator	IBM Default	Not supported

Select a Cryptographic number and then click the task push button.

View Details...
Test RN Generator
Zeroize
TKE Commands...
Crypto Type Configuration...
Zeroize All Coprocessors
Test RN Generator on All
UDX Configuration...
Refresh
Cancel
Help
https://9.152.86.25/hmc/wcl/T4b0479c6?wh=action_3fbdb9c7&action_3fbdb9c7=select(0)&timestamp=111a8c77b87#tableTop_3fbdb9c7
9.152.86.25

System z10:Cryptographic Hardware Support

New

Summary:

•Planning and installation/configuration tasks for CEX2 feature simplified by dynamic assingment of adapters/cards, domains to LPAR
➔Dynamic changes can be temporary or permanent (activation profile)
➔Candidate AP.Domains cannot be removed if they are online
➔Candidate APs cannot be added if the changed configuration would intersect with an active LPAR definition (AP.Domain)

▪New Domain Zeroize function

▪New CPACF functions with IBM System z10
➔AES 256
➔SHA 512

Dynamic Add an AP to a logical partition on z10

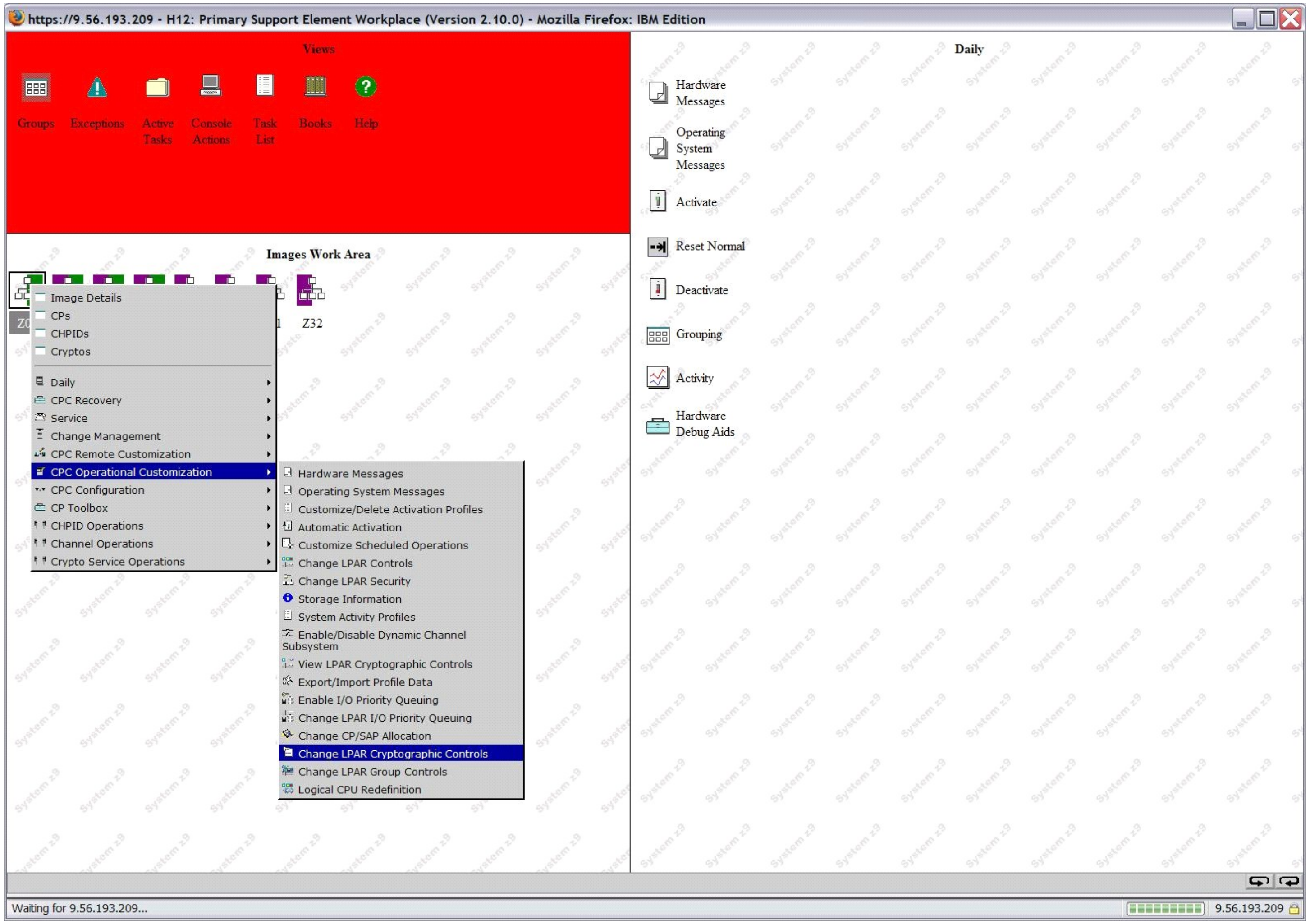
1. On the SE use ‘Change LPAR Cryptographic Controls’ Panel to **add an AP** to a Logical Partition
   - The AP appears as Standy/Stopped in the Logical Partitions work area

2. On the SE perform **Configure On** of the previously added AP for the Logical Partition
   - This is a manual action using SE panel

3. On the Operating System, issue a **Re-Sense** of the Crypto Environment, which can be manual or automatic.
   - z/OS - automatically
   - z/VSE - the APsense is an operator command, which will sense all crypto devices, including CPACF)
   - Linux - is doing an automatic sense periodically

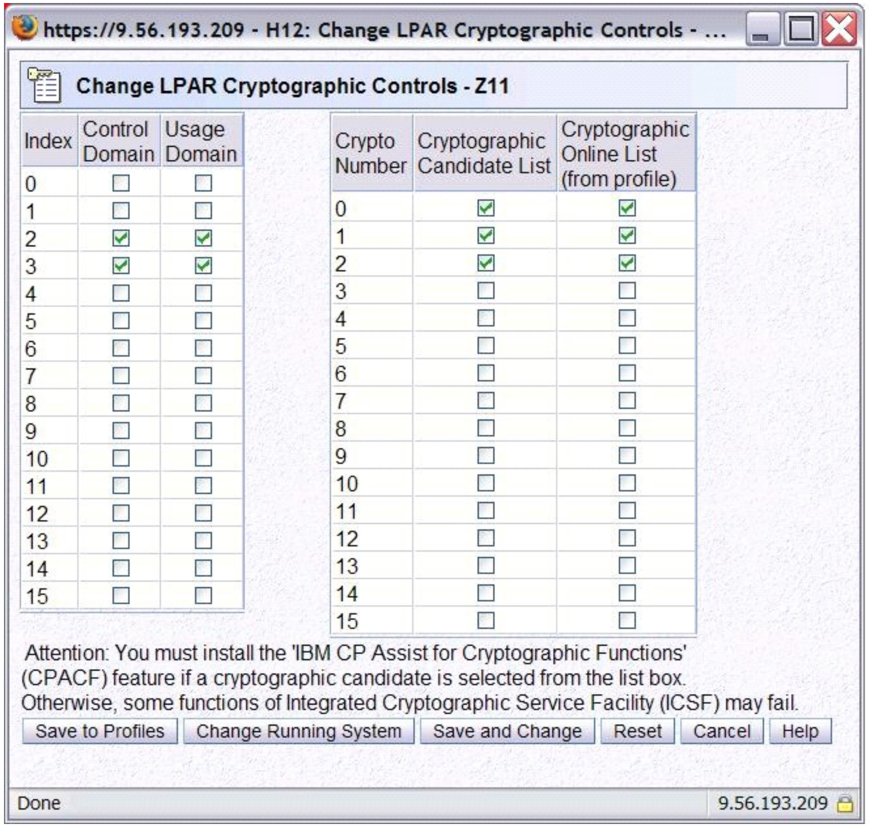
4. After this step, the new AP can be used by the Operating System in the Logical Partition

Dynamic Add an AP to a logical partition on z10


https://9.56.193.209 - H12: Primary Support Element Workplace (Version 2.10.0) - Mozilla Firefox: IBM Edition
Views
Groups
Exceptions
Active Tasks
Console Actions
Task List
Books
Help
Images Work Area
Z32
Image Details
CPs
CHPIDs
Cryptos
Daily
CPC Recovery
Service
Change Management
CPC Remote Customization
CPC Operational Customization
CPC Configuration
CP Toolbox
CHPID Operations
Channel Operations
Crypto Service Operations
Hardware Messages
Operating System Messages
Customize/Delete Activation Profiles
Automatic Activation
Customize Scheduled Operations
Change LPAR Controls
Change LPAR Security
Storage Information
System Activity Profiles
Enable/Disable Dynamic Channel Subsystem
View LPAR Cryptographic Controls
Export/Import Profile Data
Enable I/O Priority Queuing
Change LPAR I/O Priority Queuing
Change CP/SAP Allocation
Change LPAR Cryptographic Controls
Change LPAR Group Controls
Logical CPU Redefinition
Daily
Hardware Messages
Operating System Messages
Activate
Reset Normal
Deactivate
Grouping
Activity
Hardware Debug Aids
Waiting for 9.56.193.209...
9.56.193.209

On SE or via HMC / single operation mode in *Image Work Area* select the LPAR then CPC Operational Customization → Change LPAR Cryptographic Control

Dynamic Add an AP to a logical partition on z10 . . .



https://9.56.193.209 - H12: Change LPAR Cryptographic Controls - ...

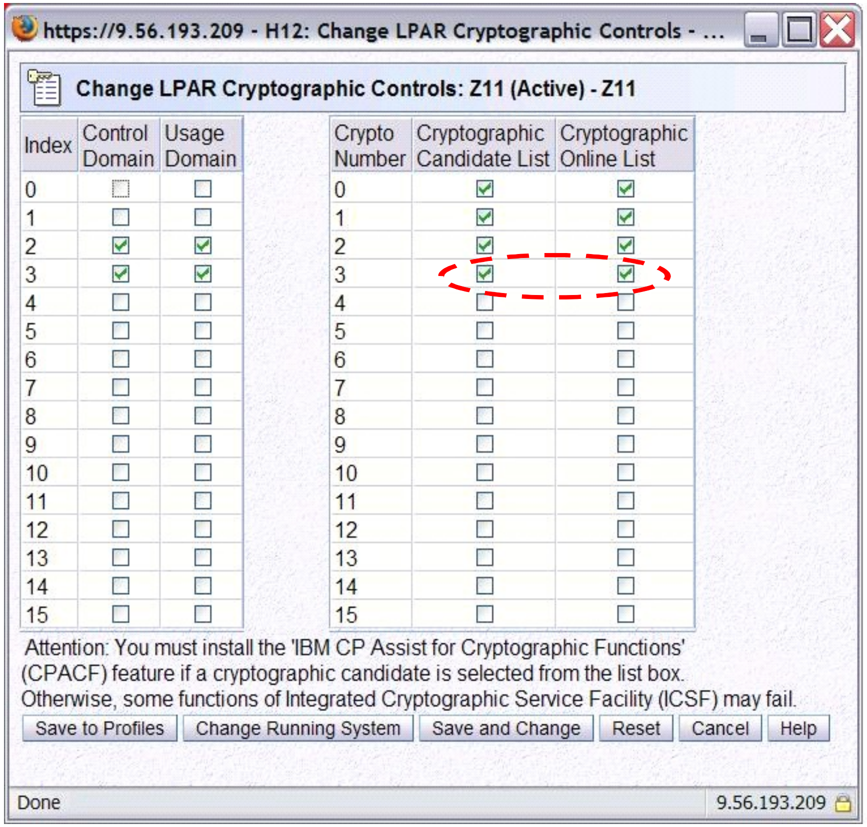
Change LPAR Cryptographic Controls - Z11

Index	Control Domain	Usage Domain
0	☐	☐
1	☐	☐
2	☑	☑
3	☑	☑
4	☐	☐
5	☐	☐
6	☐	☐
7	☐	☐
8	☐	☐
9	☐	☐
10	☐	☐
11	☐	☐
12	☐	☐
13	☐	☐
14	☐	☐
15	☐	☐

Crypto Number	Cryptographic Candidate List	Cryptographic Online List (from profile)
0	☑	☑
1	☑	☑
2	☑	☑
3	☐	☐
4	☐	☐
5	☐	☐
6	☐	☐
7	☐	☐
8	☐	☐
9	☐	☐
10	☐	☐
11	☐	☐
12	☐	☐
13	☐	☐
14	☐	☐
15	☐	☐

Attention: You must install the 'IBM CP Assist for Cryptographic Functions' (CPACF) feature if a cryptographic candidate is selected from the list box. Otherwise, some functions of Integrated Cryptographic Service Facility (ICSF) may fail.

Dynamic Add an AP to a logical partition on z10 . . .

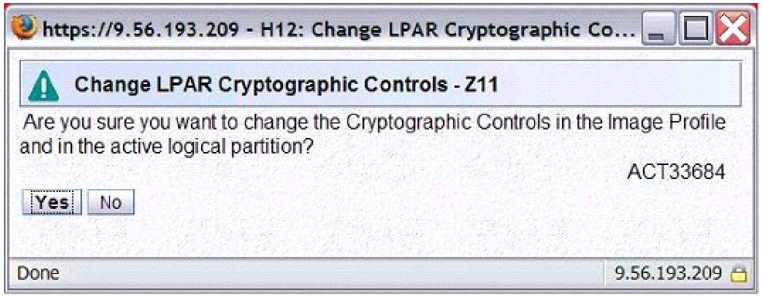


https://9.56.193.209 - H12: Change LPAR Cryptographic Controls - ...

Change LPAR Cryptographic Controls: Z11 (Active) - Z11

Index	Control Domain	Usage Domain	Crypto Number	Cryptographic Candidate List	Cryptographic Online List
0	□	□	0	☑	☑
1	□	□	1	☑	☑
2	☑	☑	2	☑	☑
3	☑	☑	3	☑	☑
4	□	□	4	□	□
5	□	□	5	□	□
6	□	□	6	□	□
7	□	□	7	□	□
8	□	□	8	□	□
9	□	□	9	□	□
10	□	□	10	□	□
11	□	□	11	□	□
12	□	□	12	□	□
13	□	□	13	□	□
14	□	□	14	□	□
15	□	□	15	□	□

Attention: You must install the 'IBM CP Assist for Cryptographic Functions' (CPACF) feature if a cryptographic candidate is selected from the list box. Otherwise, some functions of Integrated Cryptographic Service Facility (ICSF) may fail.

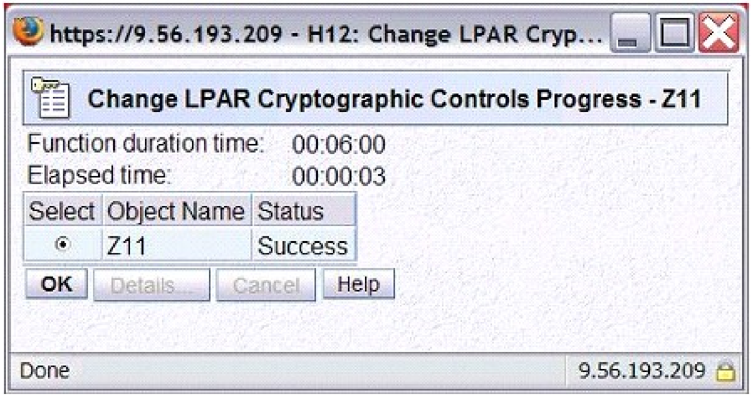


https://9.56.193.209 - H12: Change LPAR Cryptographic Co...

Change LPAR Cryptographic Controls - Z11

Are you sure you want to change the Cryptographic Controls in the Image Profile and in the active logical partition?

ACT33684



https://9.56.193.209 - H12: Change LPAR Cryp...

Change LPAR Cryptographic Controls Progress - Z11

Function duration time: 00:06:00
Elapsed time: 00:00:03

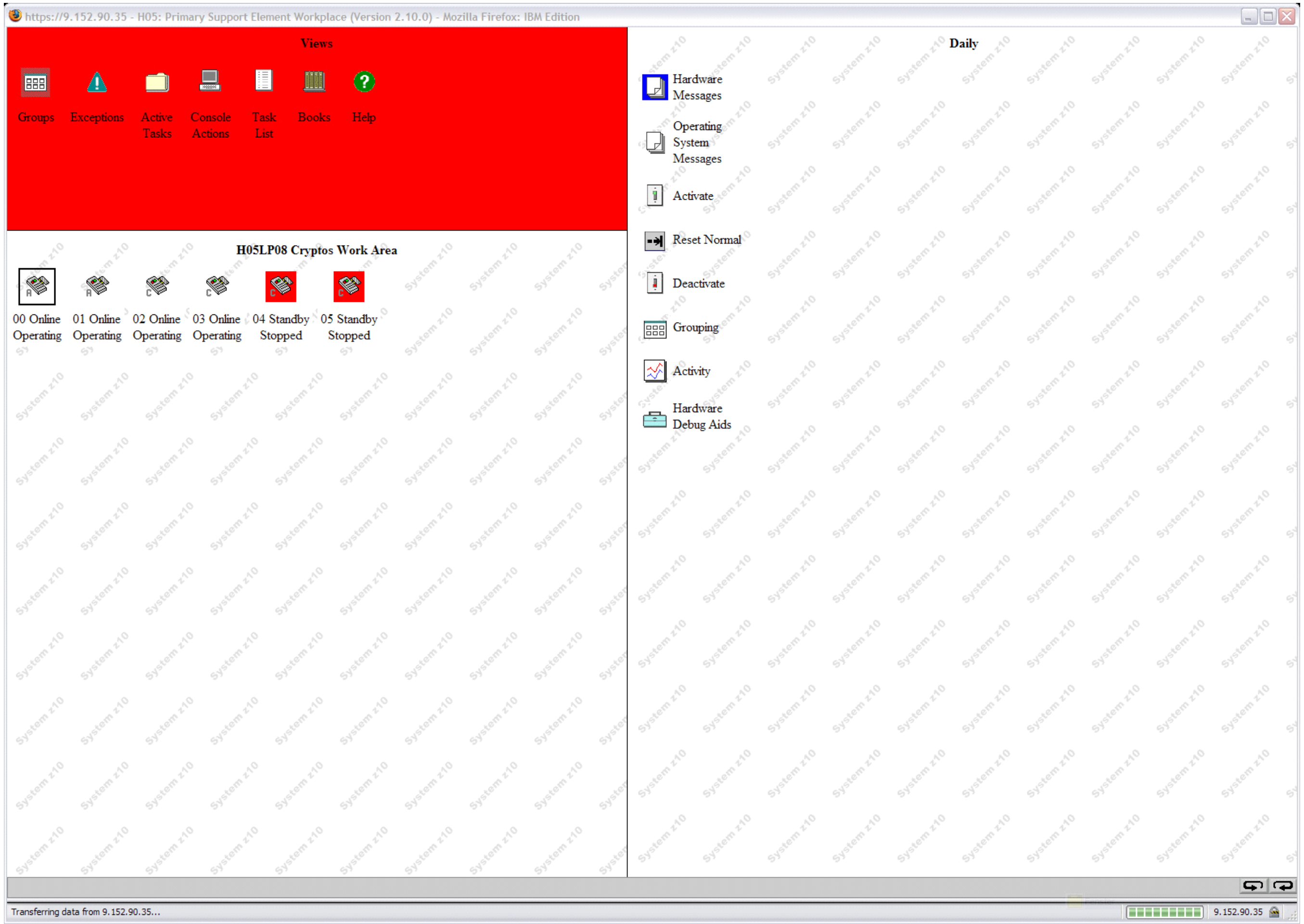
Select	Object Name	Status
●	Z11	Success

Change LPAR Cryptographic Controls allows dynamic add:

- Preparation for next LPAR activation (<Save to Profile>)
- Temporary (<Change Running System>)
- Permanent (<Save and Change>)

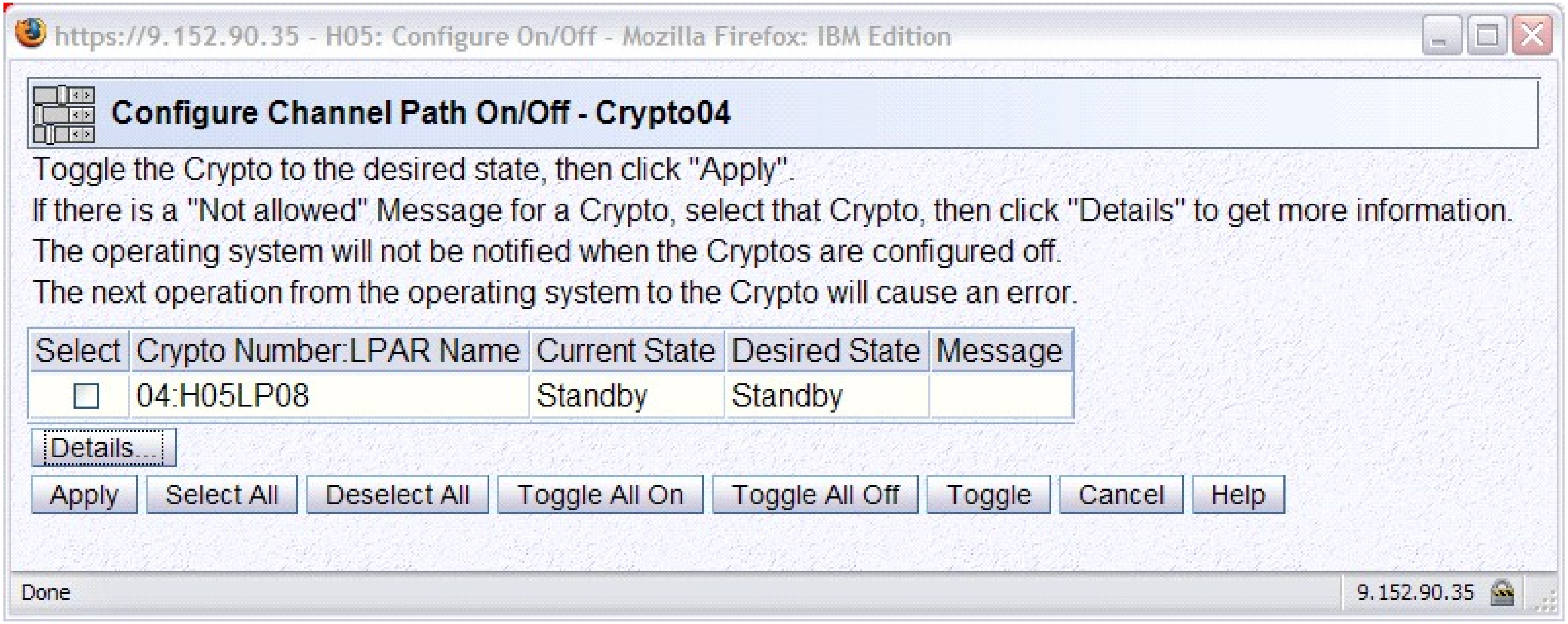
- After these dialogs AP3 shows up in *Crypto Image Area* for this LPAR as *Standby / Stopped*
- Configure On via manual operation

Dynamic Add an AP to a logical partition on z10 . . .



Configure On a previously added AP:
in Cryptos >> LPAR Crypto Work Area select corresponding AP (in Standby mode)
then *CHPID Operations -> Configure ON*

Dynamic Add an AP to a logical partition on z10 . . .



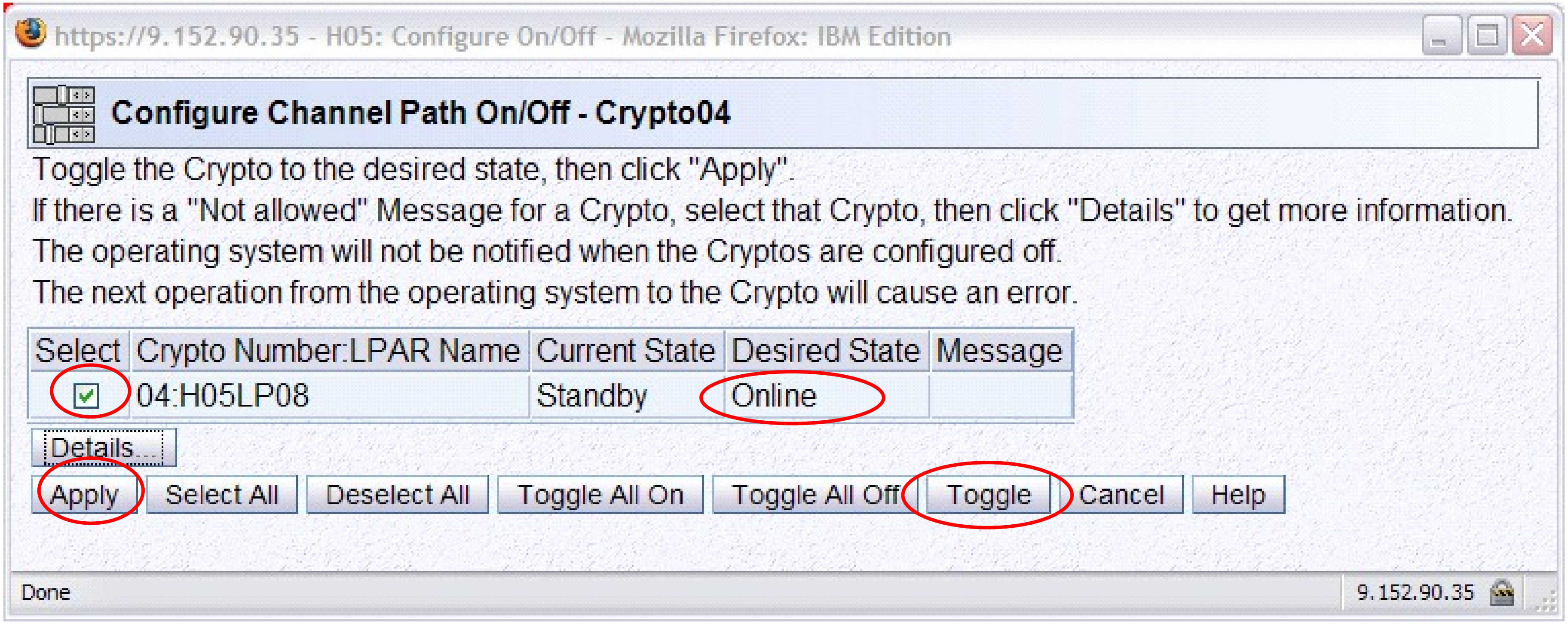
https://9.152.90.35 - H05: Configure On/Off - Mozilla Firefox: IBM Edition

Configure Channel Path On/Off - Crypto04

Toggle the Crypto to the desired state, then click "Apply".
If there is a "Not allowed" Message for a Crypto, select that Crypto, then click "Details" to get more information.
The operating system will not be notified when the Cryptos are configured off.
The next operation from the operating system to the Crypto will cause an error.

Select	Crypto Number:LPAR Name	Current State	Desired State	Message
□	04:H05LP08	Standby	Standby	

Dynamic Add an AP to a logical partition on z10 . . .



https://9.152.90.35 - H05: Configure On/Off - Mozilla Firefox: IBM Edition

Configure Channel Path On/Off - Crypto04

Toggle the Crypto to the desired state, then click "Apply".
If there is a "Not allowed" Message for a Crypto, select that Crypto, then click "Details" to get more information.
The operating system will not be notified when the Cryptos are configured off.
The next operation from the operating system to the Crypto will cause an error.

Select	Crypto Number:LPAR Name	Current State	Desired State	Message
☑	04:H05LP08	Standby	Online	

To use the „new“ AP by the Operating System running in the LPAR a Re-Sense of Crypto environment is needed – automatically or manually

- z/OS automatically
- Linux: automatic sense periodically
- z/VSE: APsense via operator command

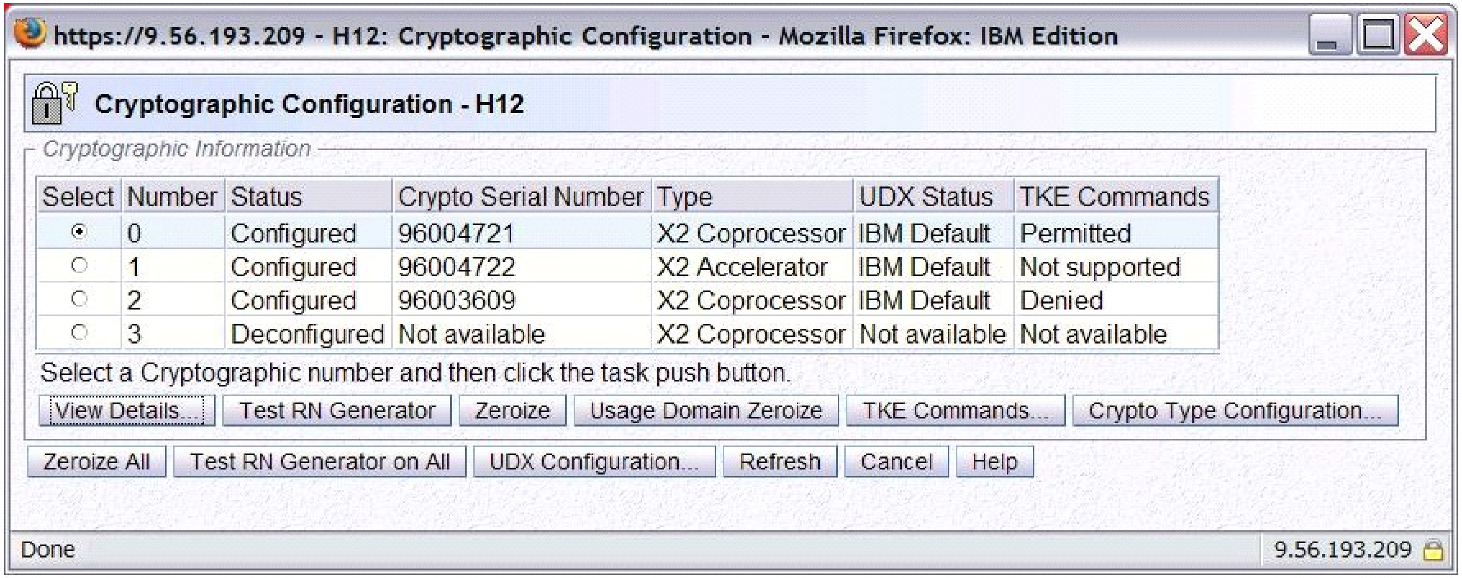
Other Dynamic Crypto changes to a logical partition on z10

- •Similar to dynamic add an AP is the remove of an AP
  - •Candidate AP.Domains can not be removed if they are onlines

- •Similar the dynamic add or remove of a Domain

- •All Dynamic changes to Cryptographic configuration cause a Security Log to be written.
- •Domain Zeroise actions are logged in Console Events and IQYYLOG

New for z10: Usage Domain Zeroize



https://9.56.193.209 - H12: Cryptographic Configuration - Mozilla Firefox: IBM Edition

Cryptographic Configuration - H12

Cryptographic Information

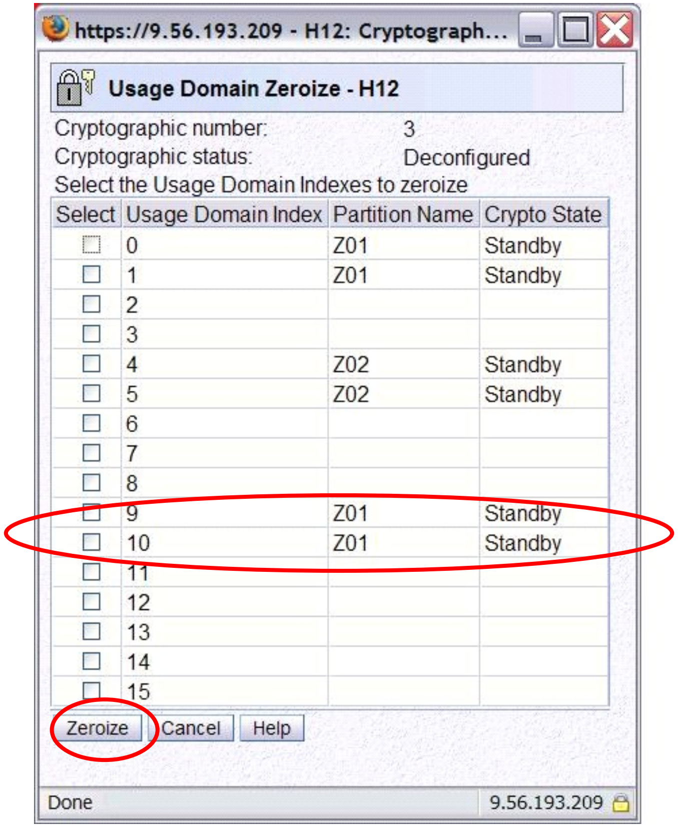
Select	Number	Status	Crypto Serial Number	Type	UDX Status	TKE Commands
●	0	Configured	96004721	X2 Coprocessor	IBM Default	Permitted
○	1	Configured	96004722	X2 Accelerator	IBM Default	Not supported
○	2	Configured	96003609	X2 Coprocessor	IBM Default	Denied
○	3	Deconfigured	Not available	X2 Coprocessor	Not available	Not available

Select a Cryptographic number and then click the task push button.

New: **Usage Domain Zeroize**

- Ability to zeroize Secrets in a Crypto on Domain basis.
- Immediate Domain Zeroize (Crypto Configured)
- Pending Domain Zeroize (Crypto Deconfigured)
  - Domain zeroize will be executed at Config On for this Crypto

New for z10: Usage Domain Zeroize . . .



Usage Domain Zeroize - H12

Cryptographic number: 3

Cryptographic status: Deconfigured

Select the Usage Domain Indexes to zeroize

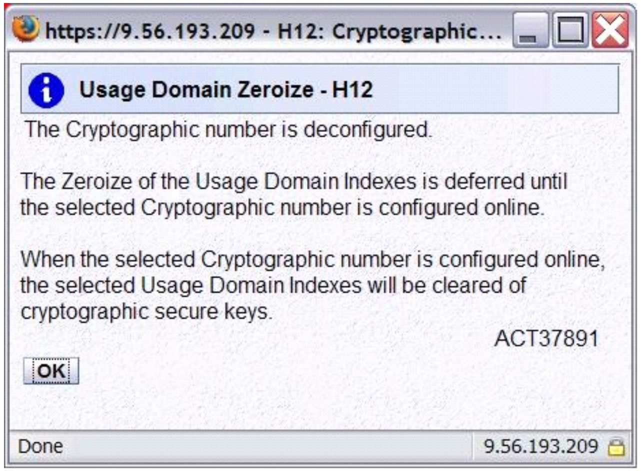
Select	Usage Domain Index	Partition Name	Crypto State
□	0	Z01	Standby
□	1	Z01	Standby
□	2		
□	3		
□	4	Z02	Standby
□	5	Z02	Standby
□	6		
□	7		
□	8		
□	9	Z01	Standby
□	10	Z01	Standby
□	11		
□	12		
□	13		
□	14		
□	15		

Example:

Issue a Usage Domain Zeroize on Domain 9 and 10 of deconfigured AP

Action will be defferred until configured online
(logged in Console Events).

New for z10: Usage Domain Zeroize . . .



Usage Domain Zeroize - H12

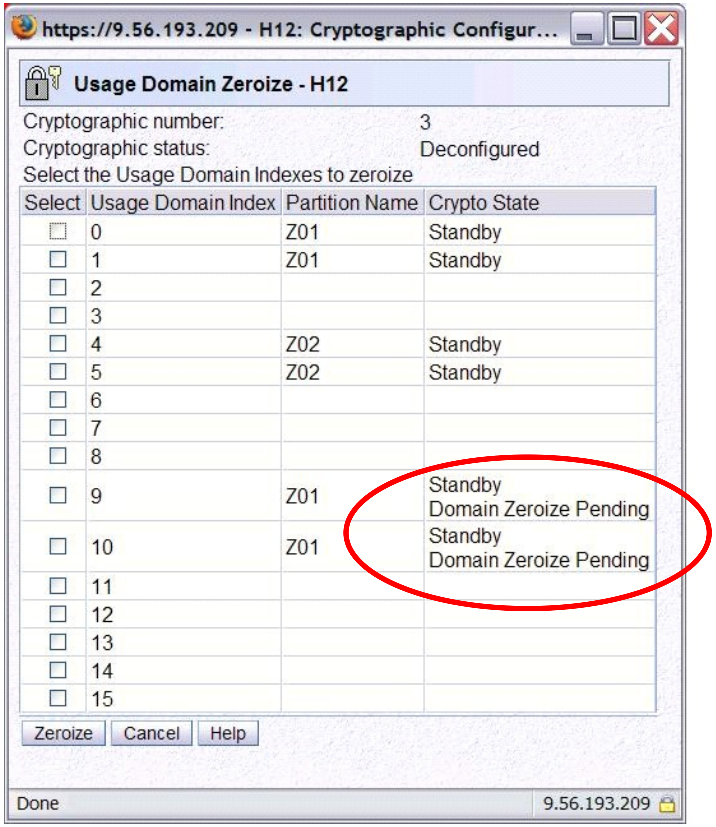
The Cryptographic number is deconfigured.

The Zeroize of the Usage Domain Indexes is deferred until the selected Cryptographic number is configured online.

When the selected Cryptographic number is configured online, the selected Usage Domain Indexes will be cleared of cryptographic secure keys.

ACT37891

New for z10: Usage Domain Zeroize . . .



Usage Domain Zeroize - H12

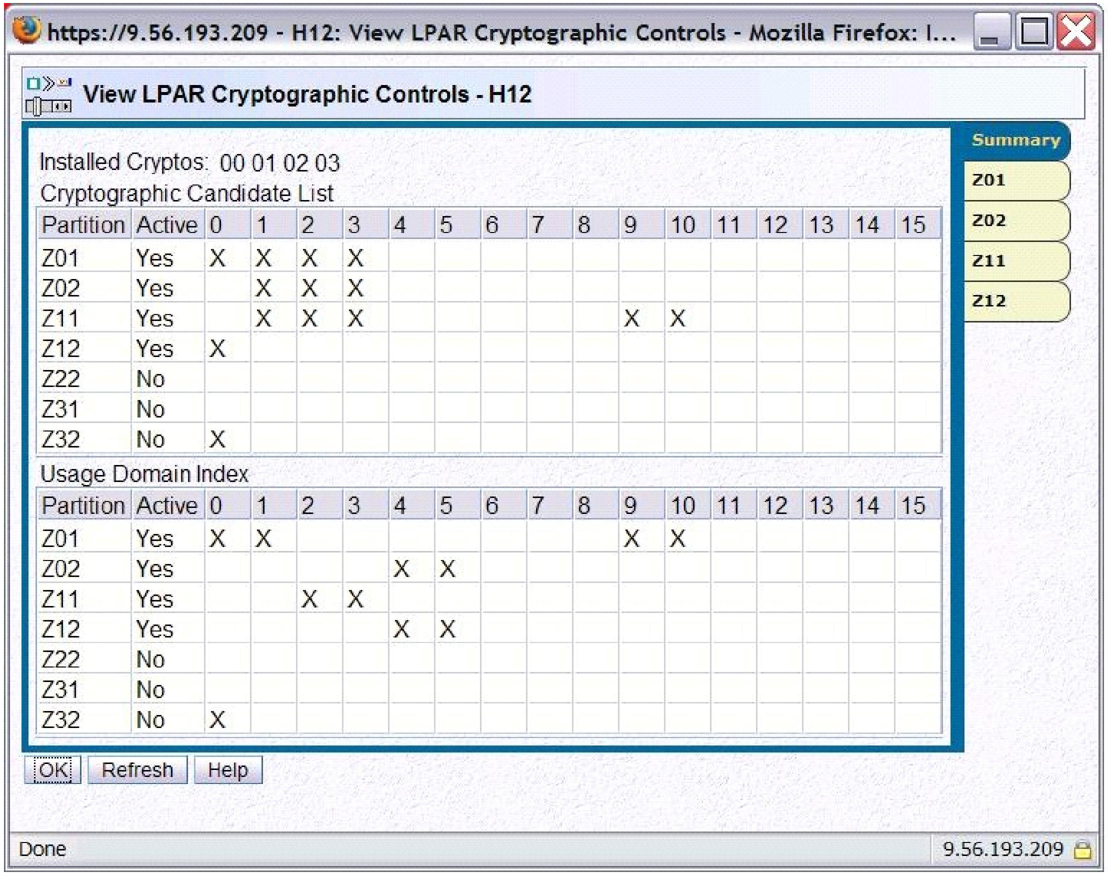
Cryptographic number: 3

Cryptographic status: Deconfigured

Select the Usage Domain Indexes to zeroize

Select	Usage Domain Index	Partition Name	Crypto State
□	0	Z01	Standby
□	1	Z01	Standby
□	2		
□	3		
□	4	Z02	Standby
□	5	Z02	Standby
□	6		
□	7		
□	8		
□	9	Z01	Standby Domain Zeroize Pending
□	10	Z01	Standby Domain Zeroize Pending
□	11		
□	12		
□	13		
□	14		
□	15		

View LPAR Cryptographic controls – new summary for z10


https://9.56.193.209 - H12: View LPAR Cryptographic Controls - Mozilla Firefox: I...
View LPAR Cryptographic Controls - H12
Summary
Z01
Z02
Z11
Z12
OK
Refresh
Help
Done
9.56.193.209

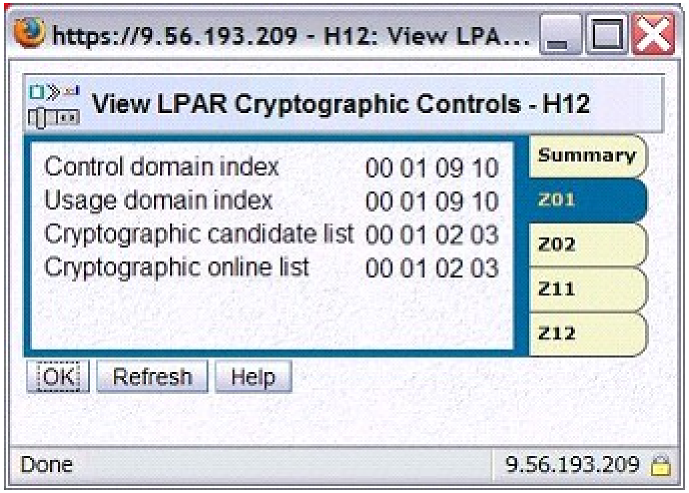
Installed Cryptos: 00 01 02 03

Cryptographic Candidate List

Partition	Active	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Z01	Yes	X	X	X	X												
Z02	Yes		X	X	X												
Z11	Yes		X	X	X						X	X					
Z12	Yes	X															
Z22	No																
Z31	No																
Z32	No	X															

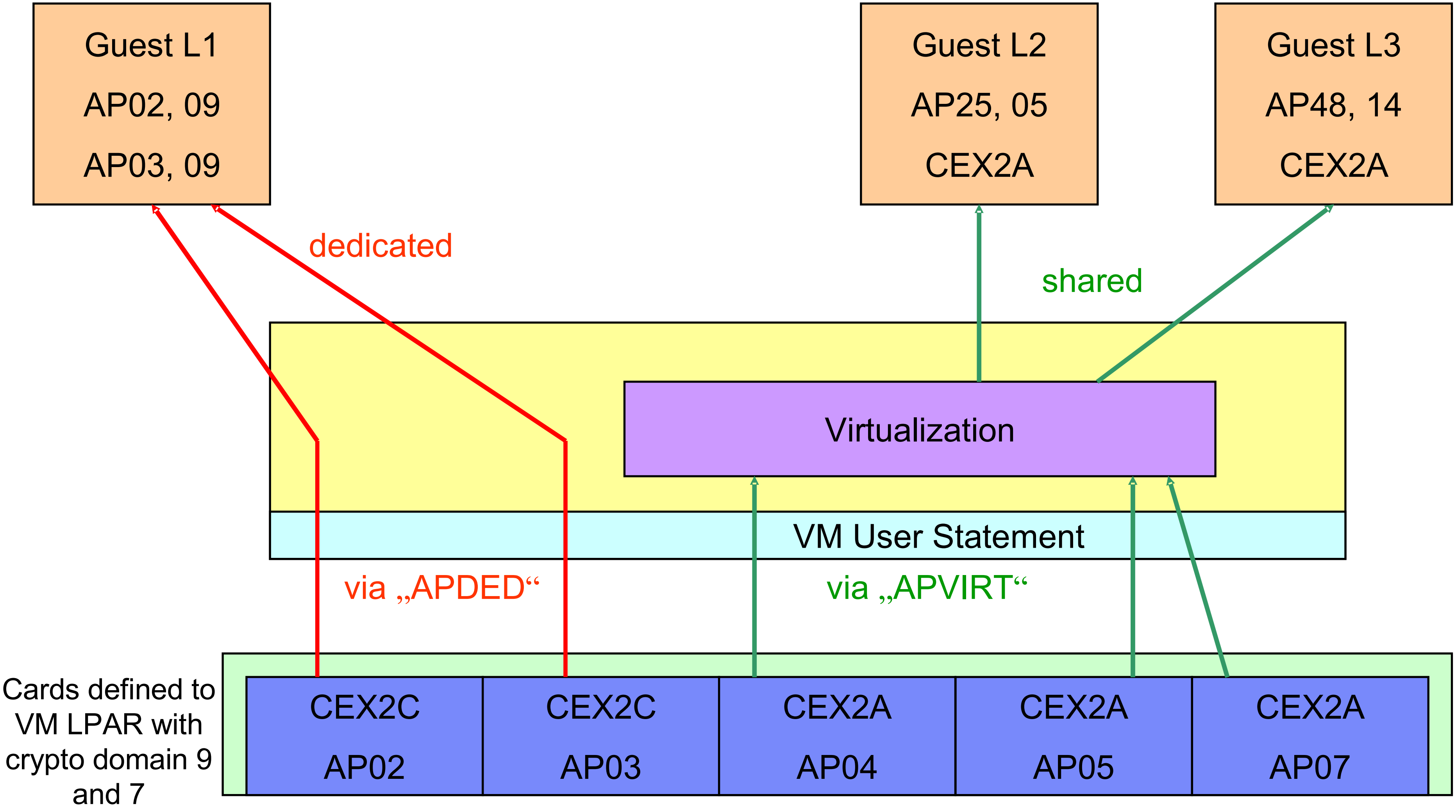
Usage Domain Index

Partition	Active	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Z01	Yes	X	X								X	X					
Z02	Yes					X	X										
Z11	Yes			X	X												
Z12	Yes					X	X										
Z22	No																
Z31	No																
Z32	No	X															


https://9.56.193.209 - H12: View LPA...
View LPAR Cryptographic Controls - H12
Summary
Z01
Z02
Z11
Z12
OK
Refresh
Help
Done
9.56.193.209

Control domain index	00 01 09 10
Usage domain index	00 01 09 10
Cryptographic candidate list	00 01 02 03
Cryptographic online list	00 01 02 03

z/VM dedicated and shared queues/adapters



z/VM dedicated and shared queues/adapters . . .

```
USER GUESTL1 xxxxxx 256M 1G G
   INCLUDE IBMDFLT
   IPL CMS
   MACH XA
   NICDEF C200 TYPE QDIO LAN SYSTEM VSWITCH
   CRYPTO DOMAIN 9 APDED 2 3
-   - - some lines not displayed - - -
USER GUESTL2 xxxxxx 256M 1G G
   INCLUDE IBMDFLT
   IPL CMS
   MACH XA
   NICDEF C200 TYPE QDIO LAN SYSTEM VSWITCH
   CRYPTO APVIRT
- - - some lines not displayed - - -
USER GUESTL3 xxxxxx 256M 1G G
   INCLUDE IBMDFLT
   IPL CMS
   MACH XA
   NICDEF C200 TYPE QDIO LAN SYSTEM VSWITCH
   CRYPTO APVIRT
- - - some lines not displayed - - -
```

z/VM: QUERY CRYPTO command

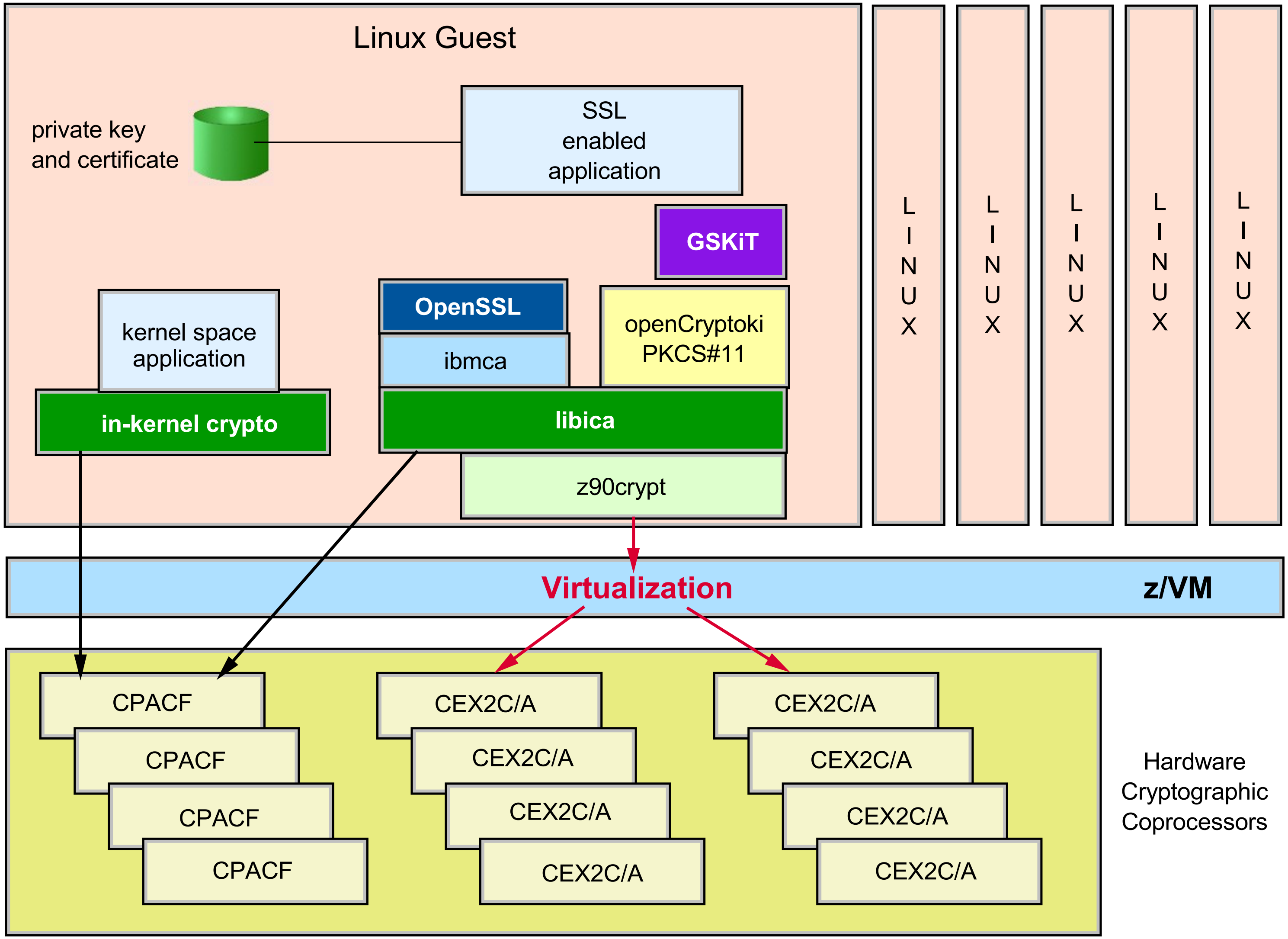
- •Displays the status of the crypto units in the processor configuration and status of the domains and AP queues
  (Cyrpto Asyn. Messages (CAM) and Direct Attached Crypto (DAD) refers to server prior to z990, z890).
- •Authorization: Privilege class A,B,C,E

```
cp q crypto
Crypto Adjunt Processor Instructions are installed

cp q crypto ap
AP00 CEX2A Queue 11 is installed
AP01 CEX2A Queue 11 is installed
AP02 CEX2C Queue 11 is superseded by CEX2A
AP02 CEX2C Queue 11 is superseded by CEX2A

cp q crypto ap
AP00 CEX2A Queue 11 is installed
AP01 CEX2A Queue 11 is installed
AP02 CEX2C Queue 11 is reserved for dedicated use
AP02 CEX2C Queue 11 is superseded by CEX2A
```

Agenda

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- Setup for cryptographic hardware support on z10
- **Access of cryptographic hardware support with Linux for System z**
- In-kernel cryptographical support
- Linux applications using HW cryptographic Support
- New tool
- Cryptographic support with Java on Linux for System z
- NSS Network Security Services
- openssh
- Secure key cryptography

Access to Cryptographic Hardware Support with Linux for System z



Hardware Cryptographic Coprocessors

Agenda

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- Setup for cryptographic hardware support on z10
- Access of cryptographic hardware support with Linux for System z
- **In-kernel cryptographical support**
- Linux applications using HW cryptographic Support
- New tool
- Cryptographic support with Java on Linux for System z
- NSS Network Security Services
- openssh
- Secure key cryptography

In-kernel crypto

- •Linux kernel version 2.6 provides a set of modules which execute encryption functions by the kernel (kernel –space).

- •These functions are built into the kernel as loadable modules.

- •IBM provides modules for specific support of System z9:
  des-s390, sha1_s390, sha256_s390, aes_s390, prng

- •You need CPACF enabled (feature 3863) to benefit from the support
  - – IF CPACF is not enabled, then automatically fall-back into software.
  - – CEX2A or CEX2C not necessary.
  - – APVIRT or APDED in CRYPTO statement of z/VM Linux user not necessary.

- •These modules are already shipped with the Linux distribution (SUSE SLES10 SP1)

- •Usage examples:
  - – IPSEC for secure communication
  - – Disk encryption with dm-crypt and LUKS (Linux Unified Key Setup)

In-kernel crypto

```
gnirss@tmcc-123-168:~> ls /lib/modules/2.6.16.46-0.12-
default/kernel/crypto/
aes.ko       crc32c.ko       michael_mic.ko  tea.ko
anubis.ko    crypto_null.ko  serpent.ko      tgr192.ko
arc4.ko      deflate.ko      sha1.ko         twofish.ko
blowfish.ko  des.ko          sha256.ko       wp512.ko
cast5.ko     khazad.ko       sha512.ko
cast6.ko     md4.ko          tcrypt.ko

gnirss@tmcc-123-168:~> ls /lib/modules/2.6.16.46-0.12-
default/kernel/arch/s390/crypto/
aes_s390.ko          des_s390.ko   sha256_s390.ko
crypt_s390_query.ko  prng.ko
des_check_key.ko     sha1_s390.ko
```

In-kernel crypto . . .

In-kernel crypto modules are loaded on request.

To use System z specific modules, add alias statements in modprobe.conf.local

```
gnirss@tmcc-123-168:~> cat /etc/modprobe.conf.local
#
# please add local extensions to this file
# ---- use hardware support for encryption for
# ---- in-kernel modules     MG 2.10.2007
alias   des     des_s390
alias   sha1    sha1_s390
alias   sha256  sha256_s390
alias   aes     aes_s390
```

To resolve dependencies and to update the definitions:

```
gnirss@tmcc-123-168:~> sudo /sbin/depmod -a
```

If general crypto modules are already loaded, use **`rmmod`** command for unloading.

Agenda

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- Setup for cryptographic hardware support on z10
- Access of cryptographic hardware support with Linux for System z
- In-kernel cryptographical support
- **Linux applications using HW cryptographic Support**
- New tool
- Cryptographic support with Java on Linux for System z
- NSS Network Security Services
- openssh
- Secure key cryptography

z90crypt device driver

- Acess to CEX2C and CEX2A for clear key encryption
- Acess to CEX2C for secure key encryption

- z90crypt supports only 1 domain

- z90crypt can select domain automatically
  - Not necessary to specify a domain for clear key

    Domain=-1 (this is the default) is used: Domain with highest number or AP devices (AP queues) is used. If multiple domains with identical (highest) number of AP devices, then domain with lowest number is used.
  - Specify a domain for secure key
- If multiple AP devices, then improved load balancing between devices

- Poll thread to reduce latency for an application while waiting for result of CEX2C or CEX2A execution.

- Poll_thread=1 system is polling for result while waiting (attention, this is CPU intensive)

- Specify domain and poll_thread during load or in /etc/sysconfig/z90crypt

- modprob, insmod, or script rcz90crypt

- Don’t forget to configure load automatically of z90crypt for boot initialization (via chkconfig z90crypt on)

z90crypt: device driver status

```
gnirss@tmcc-123-168:~> cat /proc/driver/z90crypt
zcryptversion: 2.1.0
Cryptographic domain: 1
Total device count: 1
PCICA count: 0
PCICC count: 0
PCIXCC MCL2 count: 0
PCIXCC MCL3 count: 0
CEX2C count: 1
CEX2A count: 0
requestq count: 0
pendingq count: 0
Total open handles: 1
Online devices: 1=PCICA 2=PCICC 3=PCIXCC (MCL2) 4=PCIXCC (MCL3) 5=CEX2C 6=CEX2A
      0500000000000000 0000000000000000 0000000000000000 0000000000000000

Waiting work element counts
      0000000000000000 0000000000000000 0000000000000000 0000000000000000

Per-device successfully completed request counts
  00000000 00000143 00000000 00000000 00000000 00000000 00000000 00000000
...

gnirss@tmcc-123-168:~> cat /sys/bus/ap/devices/card01/request_count
323
```

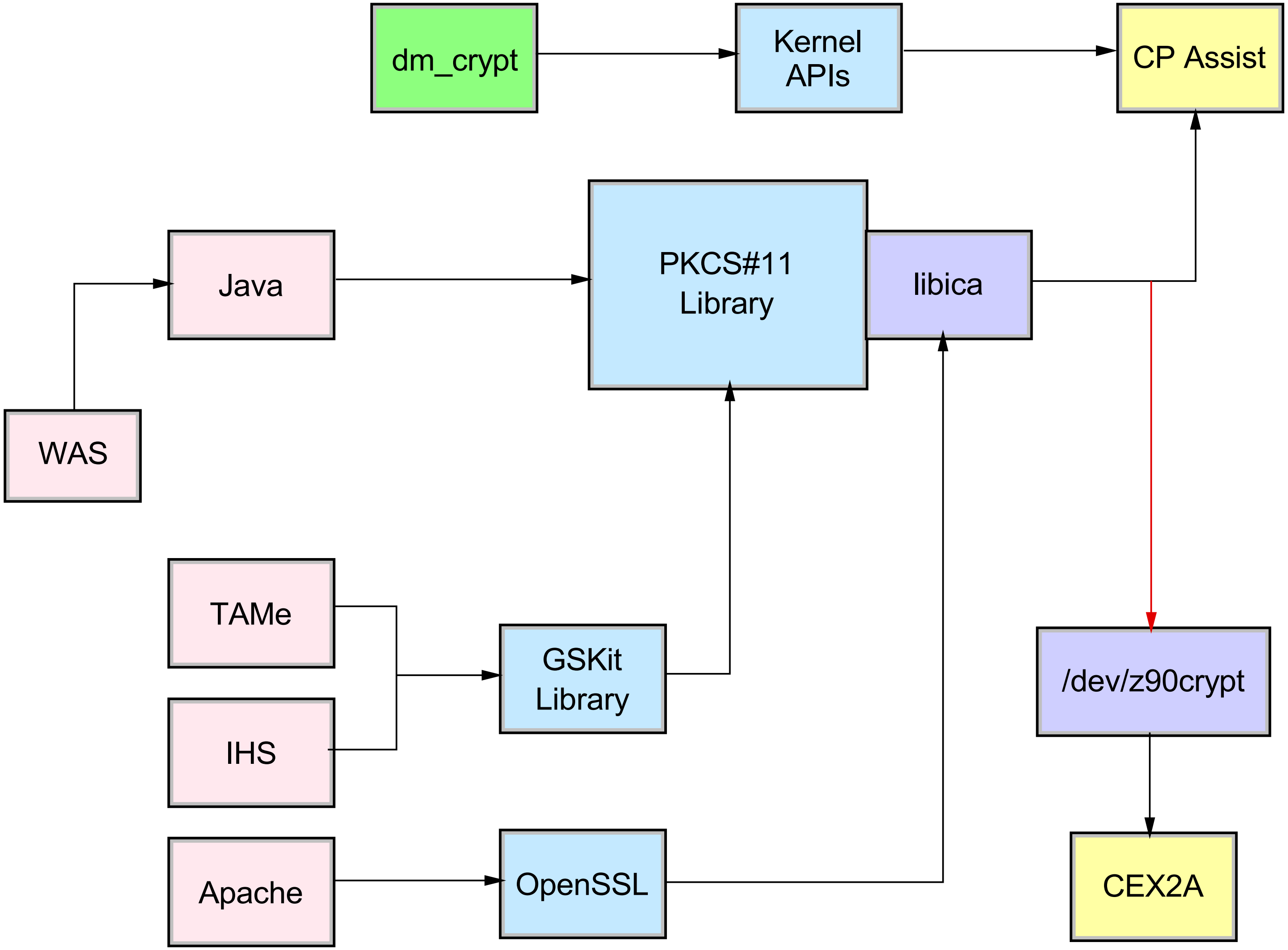
PKCS#11 - openCryptoki

openCryptoki is Open Source implementation of PKCS#11 interface to provide crypto devices that can manage and store user keys on PKCS#11 devices.
It contains:

- Slot manager daemon (**`/usr/sbin/pkcsslotd`**)
  - Controls token slots provided to application
  - Managed devices store tokens in the slot manager database
- Slot manager daemon control script (**`/etc/init.d/pkcsslotd`**)
- API for slot token dynamic link libraries (STDLLs)
  - /usr/lib/opencryptoki/libopencryptoki.so
  - /usr/lib64/opencryptoki/libopencryptoki.so
- Configuration utilities
  - /usr/sbin/pkcs11_startup
  - /usr/sbin/pkcs_slot
  - /usr/sbin/pkcsconf
  - /usr/sbin/pkcsconf64
- STDLLs plugins to the cryptographic adapters
  - /usr/lib/opencryptoki/stdll/PKCS11_ICA.so
  - /usr/lib64/opencryptoki/stdll/PKCS11_ICA.so

- For configuration of openCryptoki: see [2]

Clear key Crypto Solutions



Agenda

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- Setup for cryptographic hardware support on z10
- Access of cryptographic hardware support with Linux for System z
- In-kernel cryptographical support
- Linux applications using HW cryptographic Support
- **New tool**
- Cryptographic support with Java on Linux for System z
- NSS Network Security Services
- openssh
- Secure key cryptography

Query libica and CPACF support

New

small program icainfo to list CPACF support via libica on running system

Example: z10 without CPACF enabled

```
h05lp08:~ # icainfo
The following CP Assist for
Cryptographic Function (CPACF)
operations are supported by libica on
this system :
SHA-1:   yes
SHA-256: yes
SHA-512: yes
DES:    no
TDES-128:no
TDES-192:no
AES-128: no
AES-192: no
AES-256: no
PRNG:   no
```

Example: z10 without CPACF enabled

```
h05lp08:~ # icainfo
The following CP Assist for
Cryptographic Function (CPACF)
operations are supported by libica on
this system :
SHA-1:   yes
SHA-256: yes
SHA-512: yes
DES:    no
TDES-128:no
TDES-192:no
AES-128: no
AES-192: no
AES-256: no
PRNG:   no
```

New

Neu: Tool icastats in libica

Tool icastats
•Unterstützte kryptografische Algorithmen in libica
➔Anzahl der ausgeführten Operationen
- ➔ In Software innerhalb von libica
- ➔ Mit Hardwareunterstützung

•Noch nicht in libica Version 1.3.7 enthalten - also noch nicht in aktuellem SLES und RHEL
•Erst ab libica V2

Nach reset:

```
[root@t6329002 ~]# icastats --reset
[root@t6329002 ~]# icastats
 function | # hardware | # software
----------+------------+------------
     SHA1 |          0 |          0
   SHA224 |          0 |          0
   SHA256 |          0 |          0
   SHA384 |          0 |          0
   SHA512 |          0 |          0
   RANDOM |          0 |          0
 MOD EXPO |          0 |          0
  RSA CRT |          0 |          0
  DES ENC |          0 |          0
  DES DEC |          0 |          0
 3DES ENC |          0 |          0
 3DES DEC |          0 |          0
  AES ENC |          0 |          0
  AES DEC |          0 |          0
```

Nach Starten von Apache und SSL-Zugriff:

```
[root@t6329002 apache2]# icastats
 function | # hardware | # software
----------+------------+------------
     SHA1 |         27 |          0
   SHA224 |          0 |          0
   SHA256 |          0 |          0
   SHA384 |          0 |          0
   SHA512 |          0 |          0
   RANDOM |         10 |          0
 MOD EXPO |          6 |          0
  RSA CRT |          3 |          0
  DES ENC |          0 |          0
  DES DEC |          0 |          0
 3DES ENC |          1 |          0
 3DES DEC |         27 |          0
  AES ENC |          0 |          0
  AES DEC |          0 |          0
```

Agenda

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- Setup for cryptographic hardware support on z10
- Access of cryptographic hardware support with Linux for System z
- In-kernel cryptographical support
- Linux applications using HW cryptographic Support
- New tool
- **Cryptographic support with Java on Linux for System z**
- NSS Network Security Services
- openssh
- Secure key cryptography

Hardware Cryptographic Support with Java on Linux for System z

- IBM PKCS11 Implementation provider (IBMPKCS11Impl) uses Java Cryptographic Extension (JCE) and Java Cryptographic Architecture frameworks to seamlessly add capability to use hardware cryptography using Public Key Cryptographic Support 11 (PKCS#11) standard.

- IBMPKCS11Impl provides
  Message Digest, symmetric and asymmetric algorithms

- z90crypt loaded
- PKCS#11 (openCryptoki) configured, token generated
- Use ikeyman to generate key/certificate
- Initialize the provider IBMPKCS11Impl (using one of three methods: Java Preference method, JAAS Login Module, direct method)
  - Depending on method, adapt the provider list in java.security file
- Run Java application

Hardware Cryptographic Support with Java on Linux for System z . . .

- Important for using Hardware Crypto via Java is Crypto provider IBMPKCS11Impl
- Need an entry in java.security file (if application does not have hardcoded values)

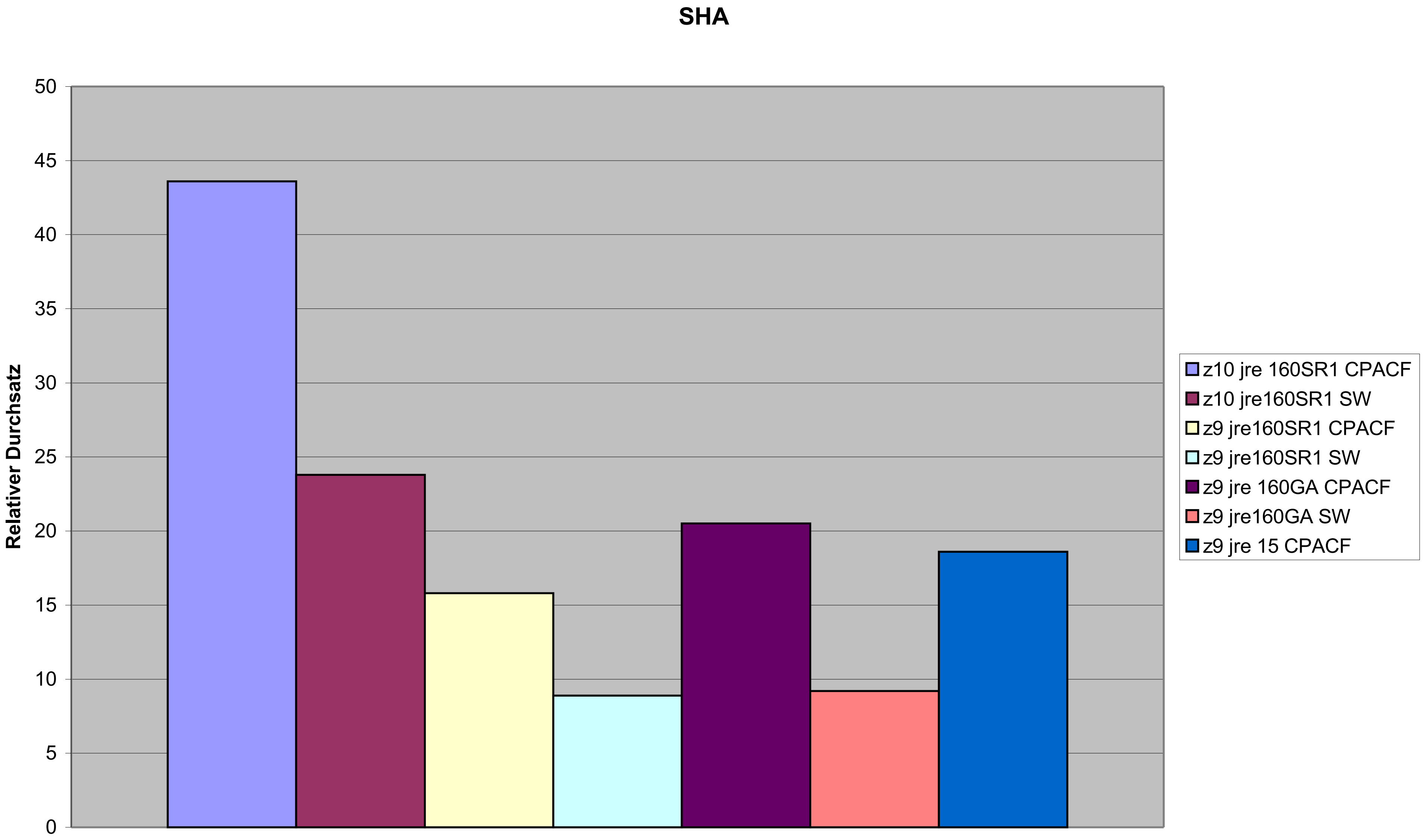
```
iic-7-229:~ # less fixedjava15/ibm-java2-s390x-
    50/jre/lib/security/java.security
. . .
#
# List of providers and their preference orders (see above):
#
security.provider.1=com.ibm.jsse2.IBMJSSEProvider2
security.provider.2=com.ibm.crypto.provider.IBMJCE
security.provider.3=com.ibm.security.jgss.IBMJGSSProvider
security.provider.4=com.ibm.security.cert.IBMCertPath
security.provider.5=com.ibm.security.sasl.IBMSASL
```

(In this default example, there is IBMPKCS11Impl missing)

Note: The followiing charts do not show official performance data

Hardware Cryptographic Support with Java on Linux for System z . . .

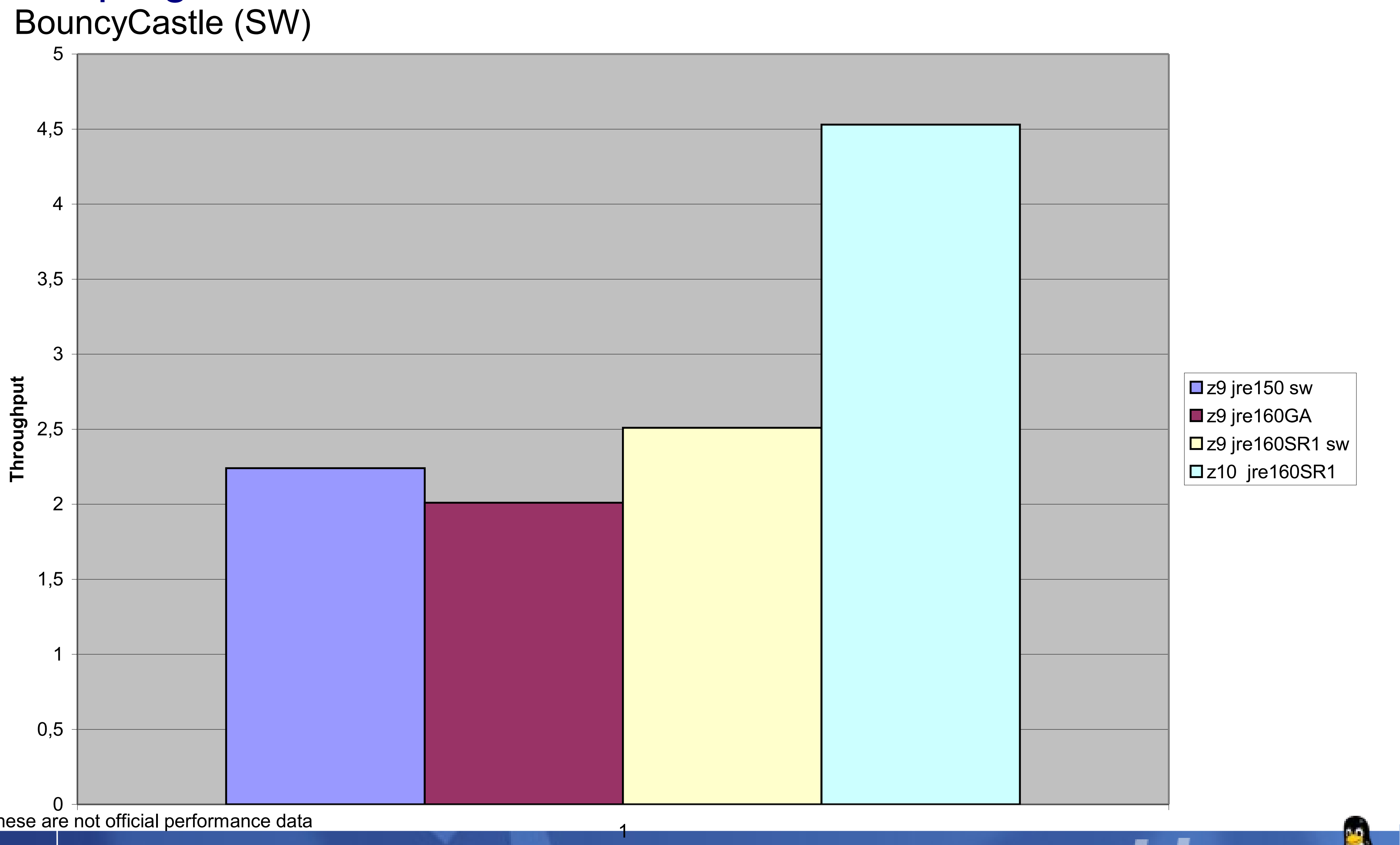
Testprogram: SHA IBMPKCS11Impl (CPACF) vs. BouncyCastle (sw)



Note: These are not official performance data

Hardware Cryptographic Support with Java on Linux for System z . . .

Testprogram: TDES

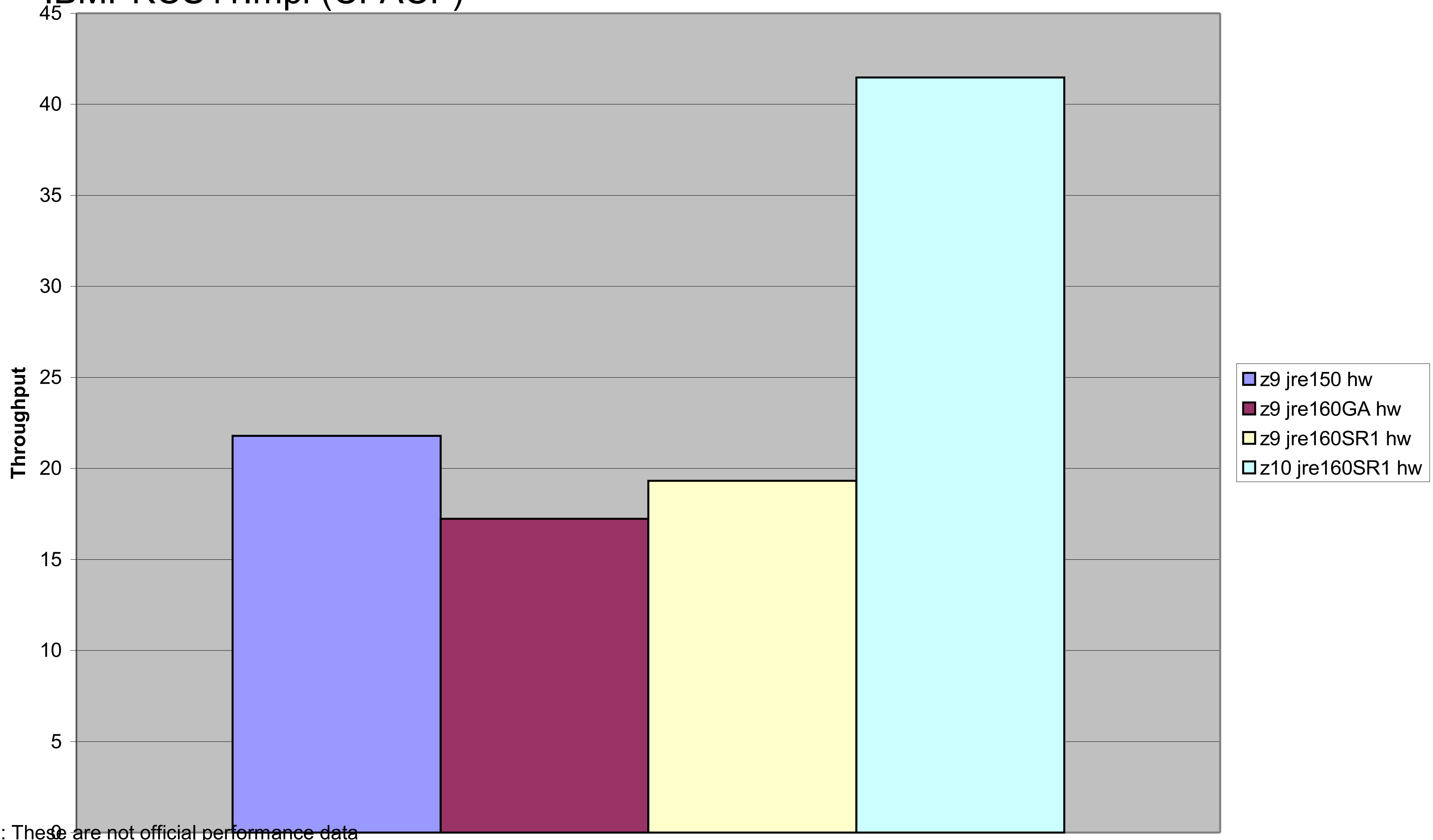


Note: These are not official performance data

Hardware Cryptographic Support with Java on Linux for System z . . .

Testprogram: TDES

IBMPKCS11Impl (CPACF)

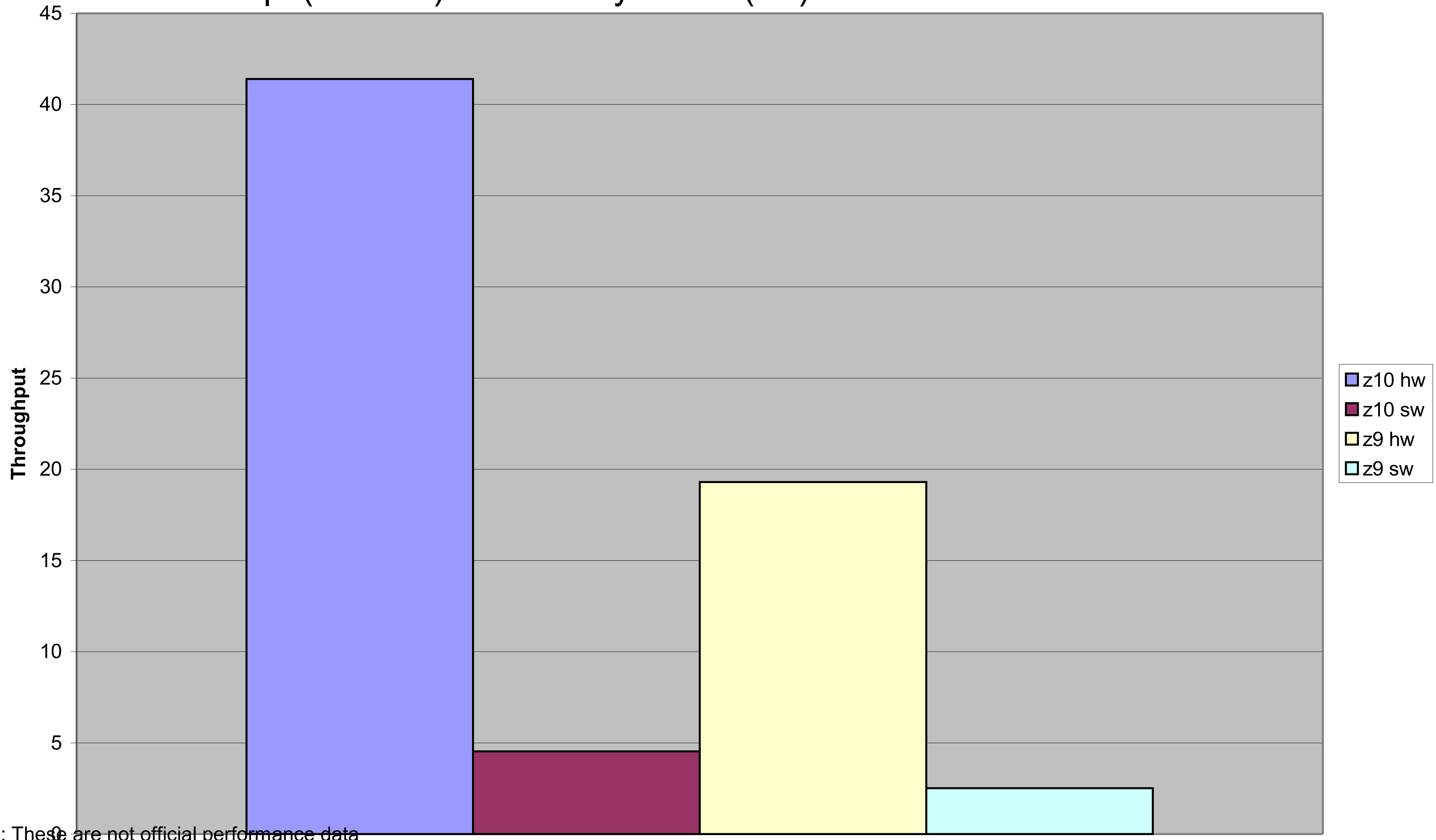


Note: These are not official performance data

Hardware Cryptographic Support with Java on Linux for System z . . .

Testprogram: TDES Throughput z9 vs. Z10

IBMPKCS11Impl (CPACF) vs. BouncyCastle (sw)



Note: These are not official performance data

Hardware Cryptographic Support with Java on Linux for System z . . .

For Comparison: Openssl speed on System z9

openssl speed -evp des-ede3-cbc (no dynamic engine loaded)

```
The 'numbers' are in 1000s of bytes per second processed.
 16 bytes   64 bytes 256 bytes 1024 bytes 8192 bytes
  . . .                                   7255.69k
```

openssl speed -evp des-ede3-cbc -engine ibmca

```
The 'numbers' are in 1000s of bytes per second processed.
 16 bytes    64 bytes  256 bytes 1024 bytes 8192 bytes
  . . .                                     247256.41k
```

Factor: ~ 34

Hardware Cryptographic Support with Java on Linux for System z . . .

For Comparison: Openssl speed on System z10

```
openssl speed -evp des-ede3-cbc (no dynamic engine loaded)
The 'numbers' are in 1000s of bytes per second processed.
 16 bytes   64 bytes 256 bytes 1024 bytes 8192 bytes
11885.17k 12119.41k 12244.39k  12208.05k  12189.70k
```

```
openssl speed -evp des-ede3-cbc -engine ibmca
The 'numbers' are in 1000s of bytes per second processed.
 16 bytes    64 bytes   256 bytes 1024 bytes 8192 bytes
72472.47k 168270.05k 259305.90k 298243.07k 310727.19k
```

Factor: ~ 26

Hardware Cryptographic Support with Java on Linux for System z . . .

Results from testprogram:

- **(Decrypt faster than encrypt: caching)**

- **Testprogram with java 1.5.0 is faster than with 1.6.0GA**
  - Several Java problems in 1.6.0GA level (some APARs): Quality as well as performance issues

- **Testprogram with java 1.6.0SR1 is faster than with 1.5.0**

- **If CPACF is used for TDES, then time for encryption part of testprogram is similar in all java versions.**

- **Testprogram shows factor 8-10 performance improvements with CPACF for TDES compared to open source Bouncy Castle crypto provider (software encryption)**

Agenda

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- Setup for cryptographic hardware support on z10
- Access of cryptographic hardware support with Linux for System z
- In-kernel cryptographical support
- Linux applications using HW cryptographic Support
- New tool
- Cryptographic support with Java on Linux for System z
- **NSS Network Security Services**
- openssh
- Secure key cryptography

NSS

NSS steht hier nicht für

…
Name Service Switch
National Security Strategy
National Security Service
National Service scheme
einen Längstwellensender bei Annapolis
Novell Storage Services
…

sondern für

Network Security Services

Network Security Services (NSS)

NSS originated from libraries developped by Netscape

NSS comprises a set of libraries for cross platform development of security-enabled client and server applications

NSS includes a open source implementation of SSL, TLS, S/MIME

AOL, RedHat, Sun Microsystems and other companies use NSS

- Mozilla client products (incl. Firefox, Thunderbird, SeaMonkey)
- AOL Communicator and AOL Instant Messanger (AIM)
- Open Source apps (Evolution, Pidgin, OpenOffice.org 2.0)
- RedHat: Directory Server, Certificate System, mod_nss for Apache
- SUN: Sun Java Enterprise System, (incl. Sun Java System Web Server, Sun Java System Directory Server, Sun Java System Portal Server, Sun Java System Messaging Server, and Sun Java System Application Server)

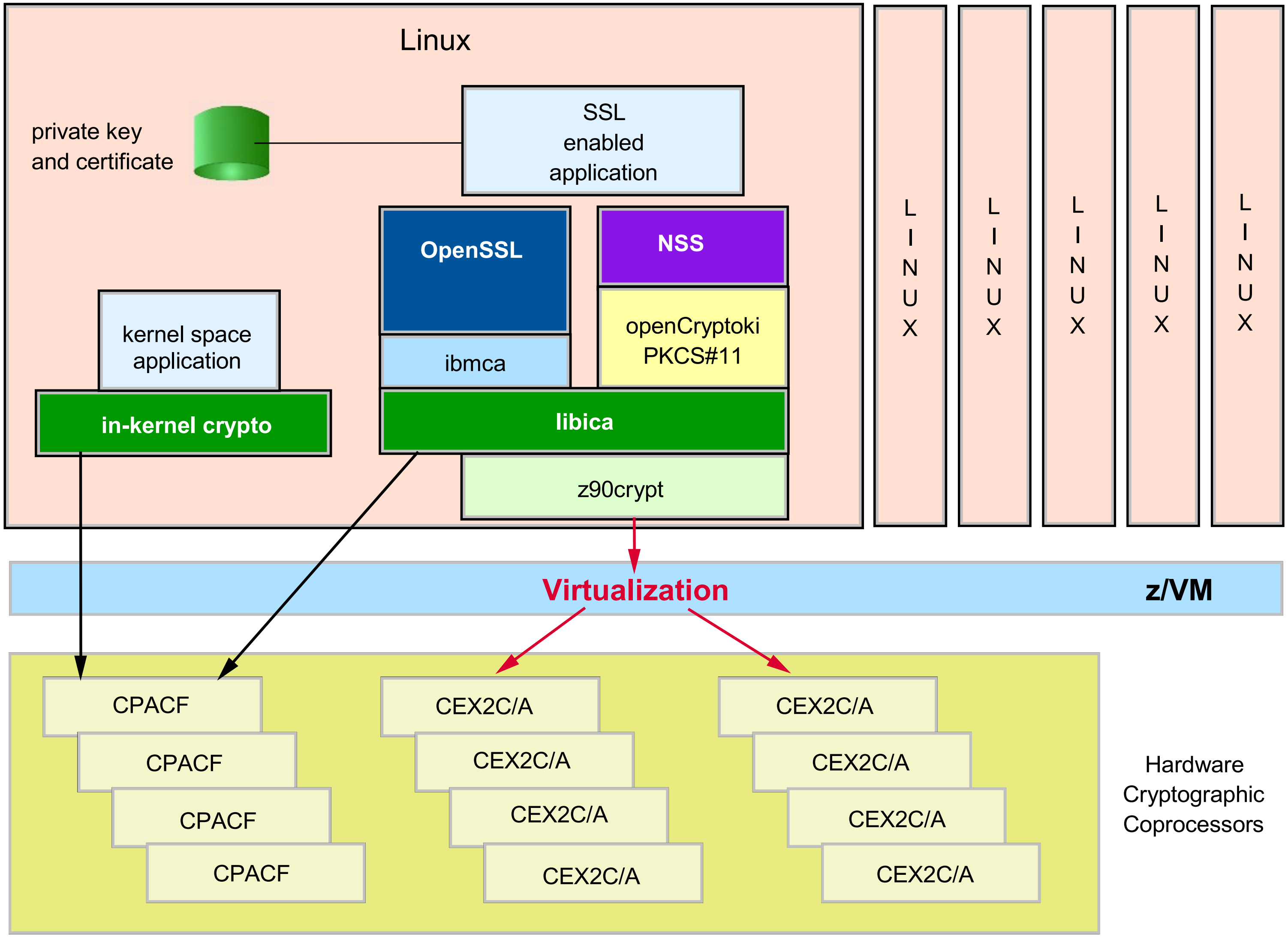
Network Security Services (NSS). . .

NSS supports:

- SSL v2 and v3
- TLS
- PKCS#1, PKCS#3, PKCS#5, PKCS#7, PKCS#8, PKCS#9, PKCS#10, PKCS#12
- PKCS#11: RSA standard that governs communication with cryptographic tokens (such as hardware accelerators and smart cards) and permits application independence from specific algorithms and implementations.

NSS software cryptographic module has been validated for FIPS 140 conformance at sec level 1 and 2 (1997, 1999, 2002)

Network Security Services (NSS). . .



Hardware
Cryptographic
Coprocessors

First Comparison of Apache on Linux for System z: NSS vs. OpenSSL

The following information is a result of

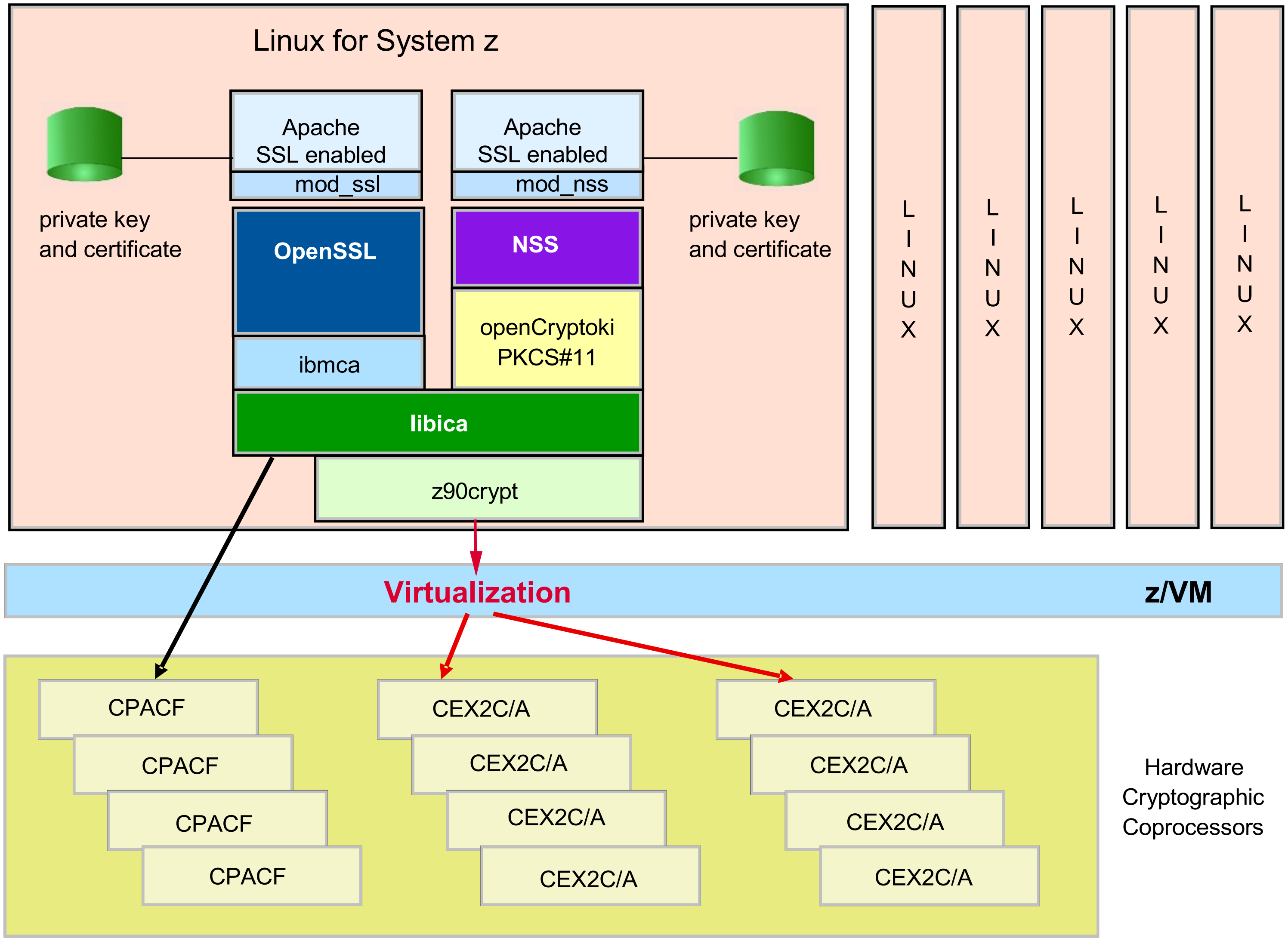
Stefan Kirchner
Westfälische Wilhelms-Universität Münster

Our best compliments belong to

PD Dr. Markus Borschbach
Rob Crittenden (Red Hat, Inc.)

and to the following IBM collegues
Rajiv Andrade
David Sadler
Thomas Weber
Arthur Winterling
Holger Wolf
Kent E. Yoder

First Comparison of Apache on Linux for System z: NSS vs. OpenSSL . . .



First Comparison of Apache on Linux for System z: NSS vs. OpenSSL . . .

Umgebung für **funktionalen Test**: RHEL und z10 mit CEX2C

NSS compile, da nicht Bestandteil von zDistro (RHEL und SLES)

NSS nutzt PKCS#11 – hat eigenes Default Modul - SW-only

Testscripte zur Funktionalität – allgemein: ok

PKCS#11 bzw. openCryptoki für HW Zugriff bereitstellen
(Konfiguration: siehe [2])

Testscripte zur RSA Test – Nachweis des Zugriff auf CEX2A/CEX2C
via z90crypt:

```
cat /proc/driver/z90crypt
cat /sys/bus/ap/devices/card#/request_count
```

Testscripte zu TDES, AES – Zugriff auf CPACF
via icastats (Für diesen Test wurde libica ausgetauscht!)

Ergebnis
NSS: CEX2 für RSA und CPACF für AES, TDES und SHA nutzbar.

First Comparison of Apache on Linux for System z: NSS vs. OpenSSL . . .

Apache für OpenSSL mit HW Unterstützung konfigurieren:

Apache mit NSS (mod_nss) compilSetup für SSL (mod_ssl)
Dann config Statement in ssl-global.conf file:

```
          SSLCryptoDevice ibmca
```

SSLCipherSuite wählen, die RSA, SHA TDES od. AES enthält (SSLCipherSuite statement in vhost-ssl.conf).

Ergebnis
Nachweis via z90crypt Informationen oder auch via icastats: Für RSA wird CEX2 gegenutzt
Nachweis via icastats: Für TDES, SHA, AES wird CPACF genutzt.

Note: To be able to use icastats, we exchanged libica

First Comparison of Apache on Linux for System z: NSS vs. OpenSSL . . .

Apache für NSS mit HW Unterstützung konfigurieren:

Setup für SSL (modd_sll mit mod_nss austauschen) – compile, da NSS nicht Bestandteil der z-Distribution

Config statements anpassen für Nutzung v. OpenCryptoki Token)

Cipher Suite wählen, die RSA, SHA, TDES od. AES enthält (NSSCipherSuite statement in nss. conf).

Ergebnis
Nachweis via z90crypt Informationen oder auch via icastats:

- Für RSA wird CEX2 genutzt

Nachweis via icastats:

- Beim Starten des Apaches sieht man ein paar wenige Requests für TDES, die CPACF nutzen.
- Beim Abruf von SSL geschützten Seiten wird weder für SHA, TDES noch AES die CPACF genutzt!!!

Note: To be able to use icastats, we exchanged libica

First Comparison of Apache on Linux for System z: NSS vs. OpenSSL . . .

„Performance“/Durchsatzvergleich zwischen Apache mit OpenSSL und Apache mit NSS

Testumgebung für Vergleich:
IBM System z9 mit CEX2A
SUSE Linux SLES 10 SP2.

Gewählte Cipher Suites:

- RSA mit RC4 und MD5

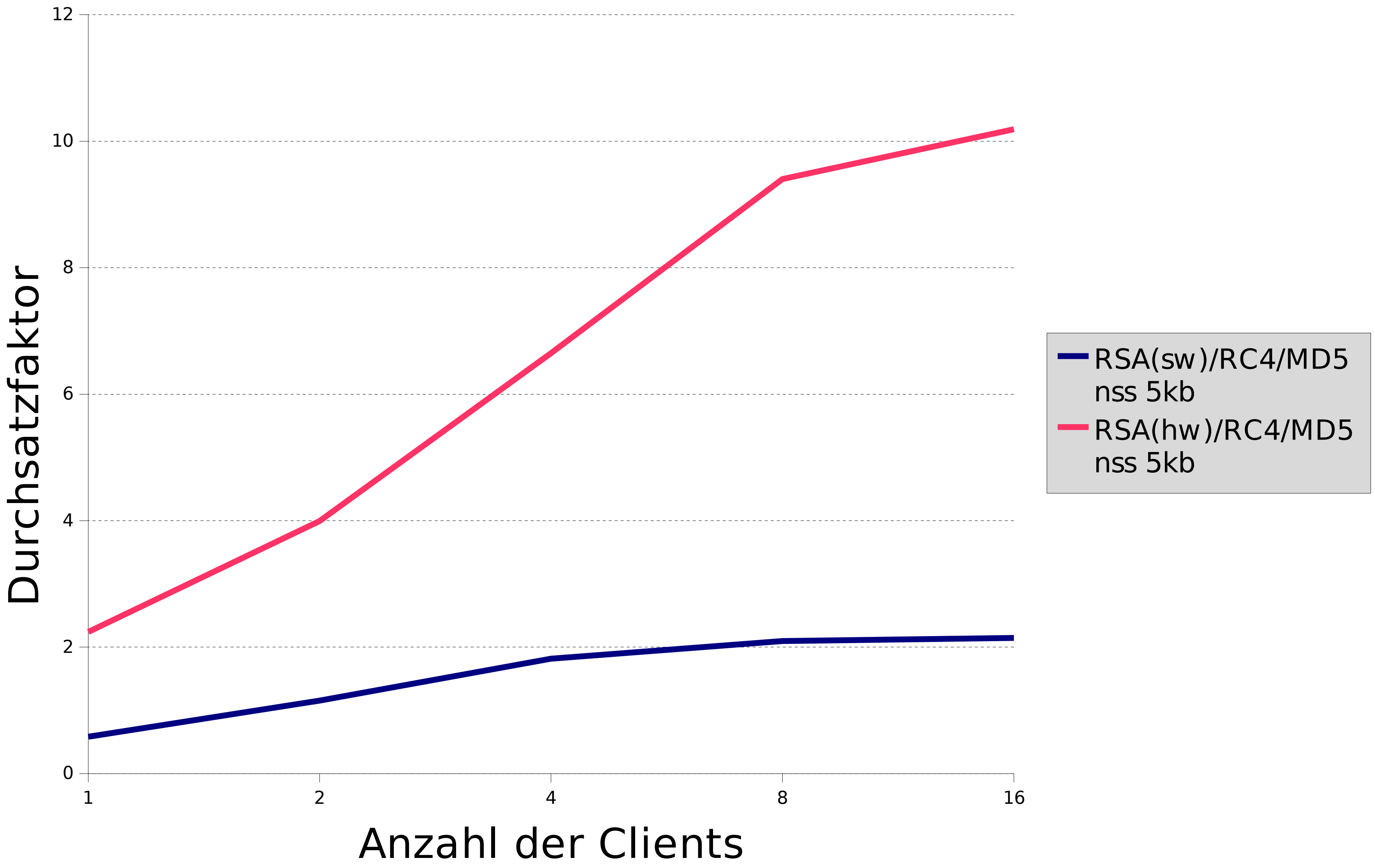
Also: RSA mit HW Unterstützung möglich und Rest rein in Software
Datengröße 5 kb – Annahme, Testcase entspricht Szenario mit vielen kurzen SSL Sssions bei wenig Datenverkehr.

- RSA mit AES-128 und SHA

Also: RSA mit Hardware Unterstützung möglich (NSS und OpenSSL)
Also: AES, SHA keine Hardware Unterstützung für NSS
Also: AES, SHA mit CPACF Unterstützung für OpenSSL
Datengröße 100 kb – Annahme: Testcase entspricht Szenario mit einem etwas höheren Nutzerdatenvolumen.

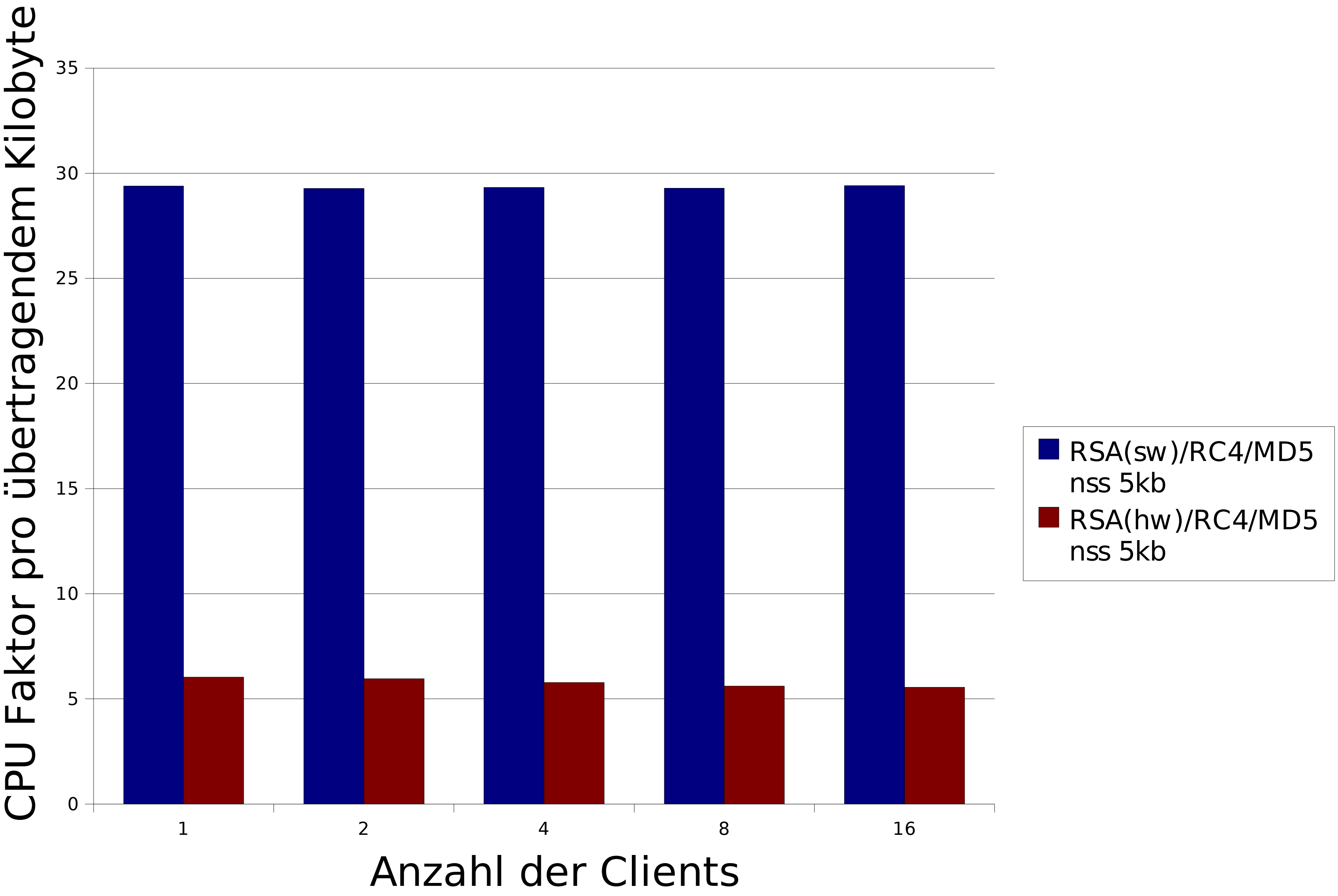
Variable Zahl von Lasttreiber (Clients), die SSL Handshake durchführen und dann geschützte Daten von Apache holen.

Apache mit mod_nss: HW ↔ SW für RSA/RC4/MD5



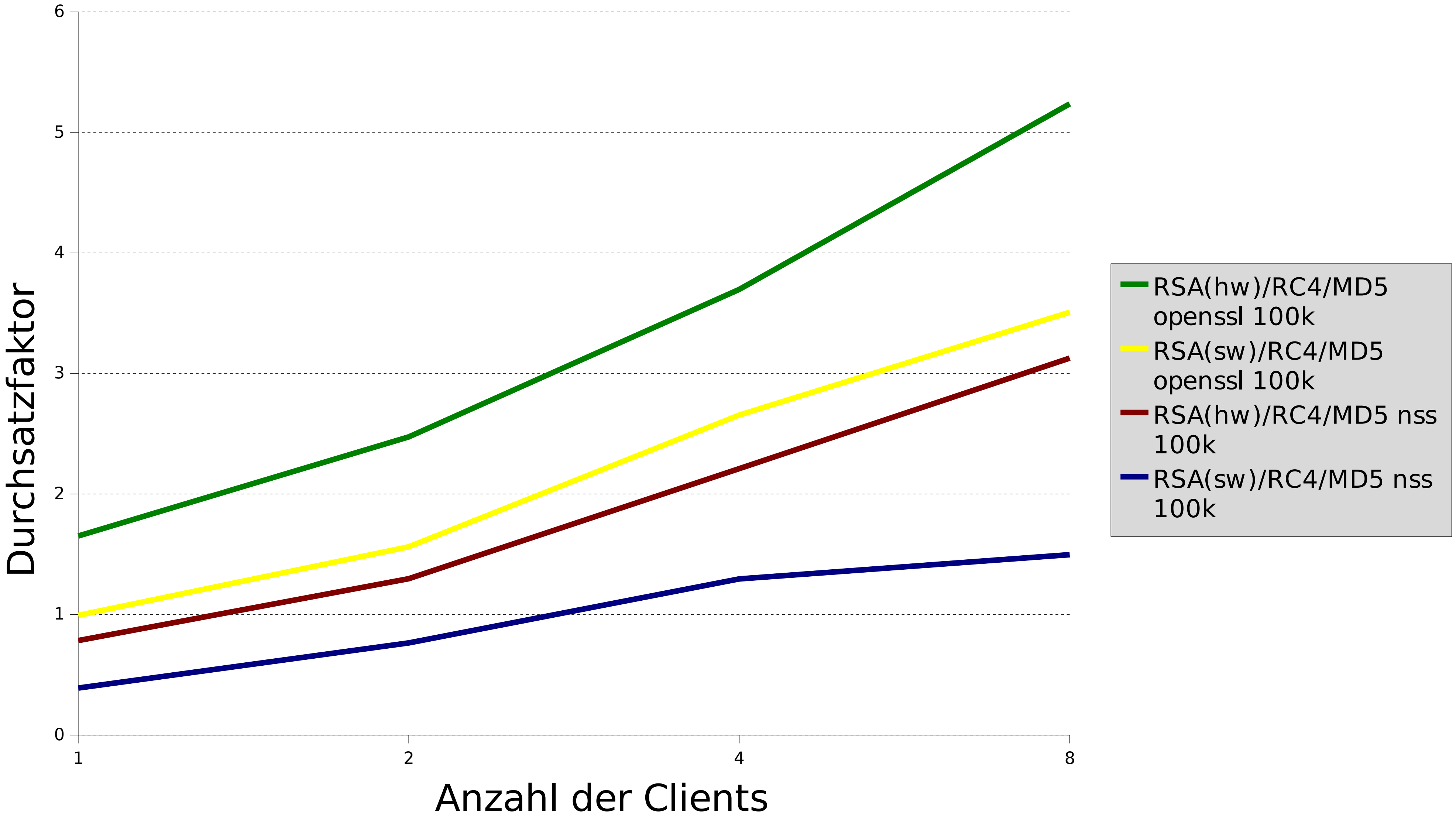
Note: These are not official performance data

Apache mit mod_nss: HW ↔ SW für RSA/RC4/MD5



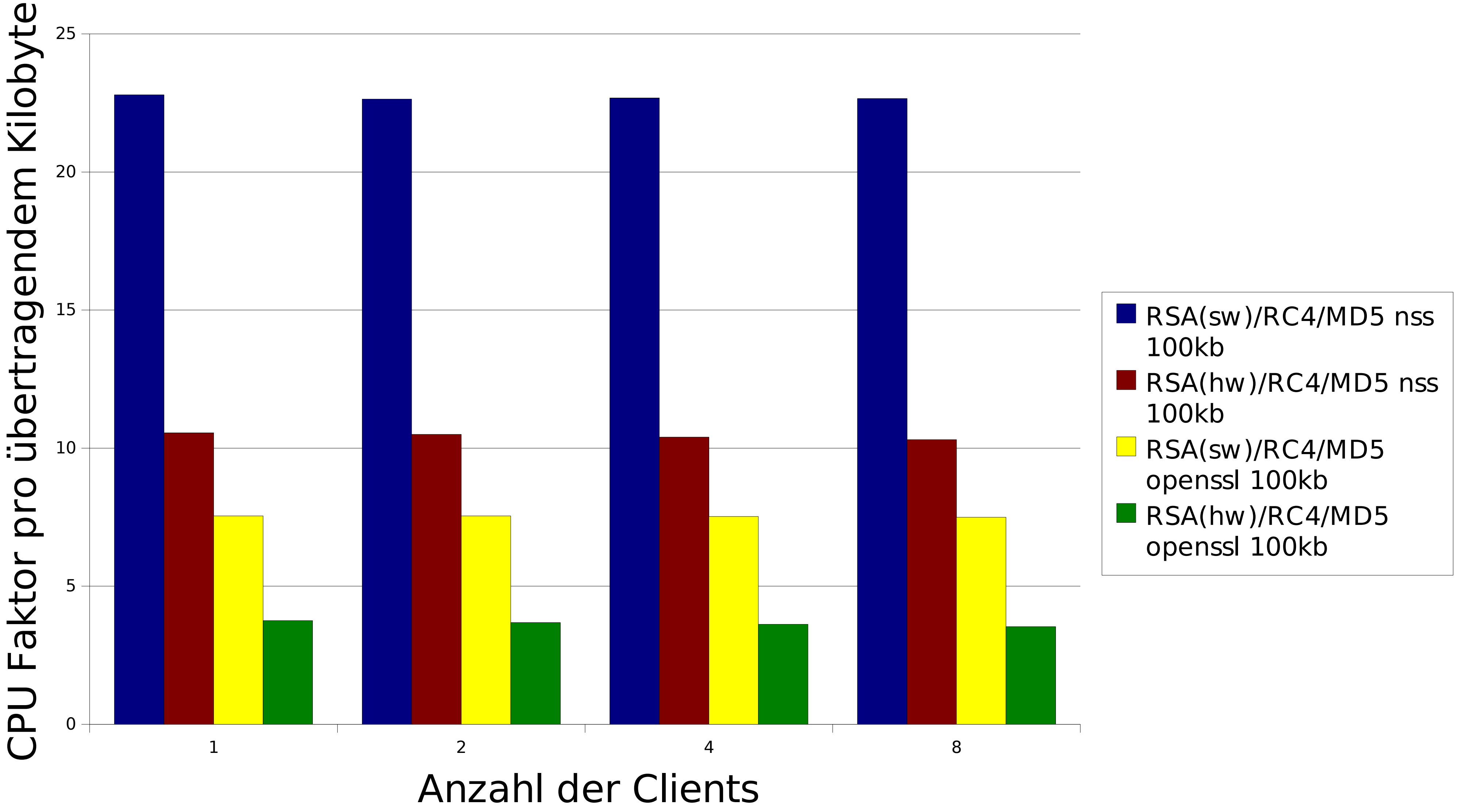
Note: These are not official performance data

Apache mit mod_nss ↔ Apache mit mod_ssl für RSA/RC4/MD5



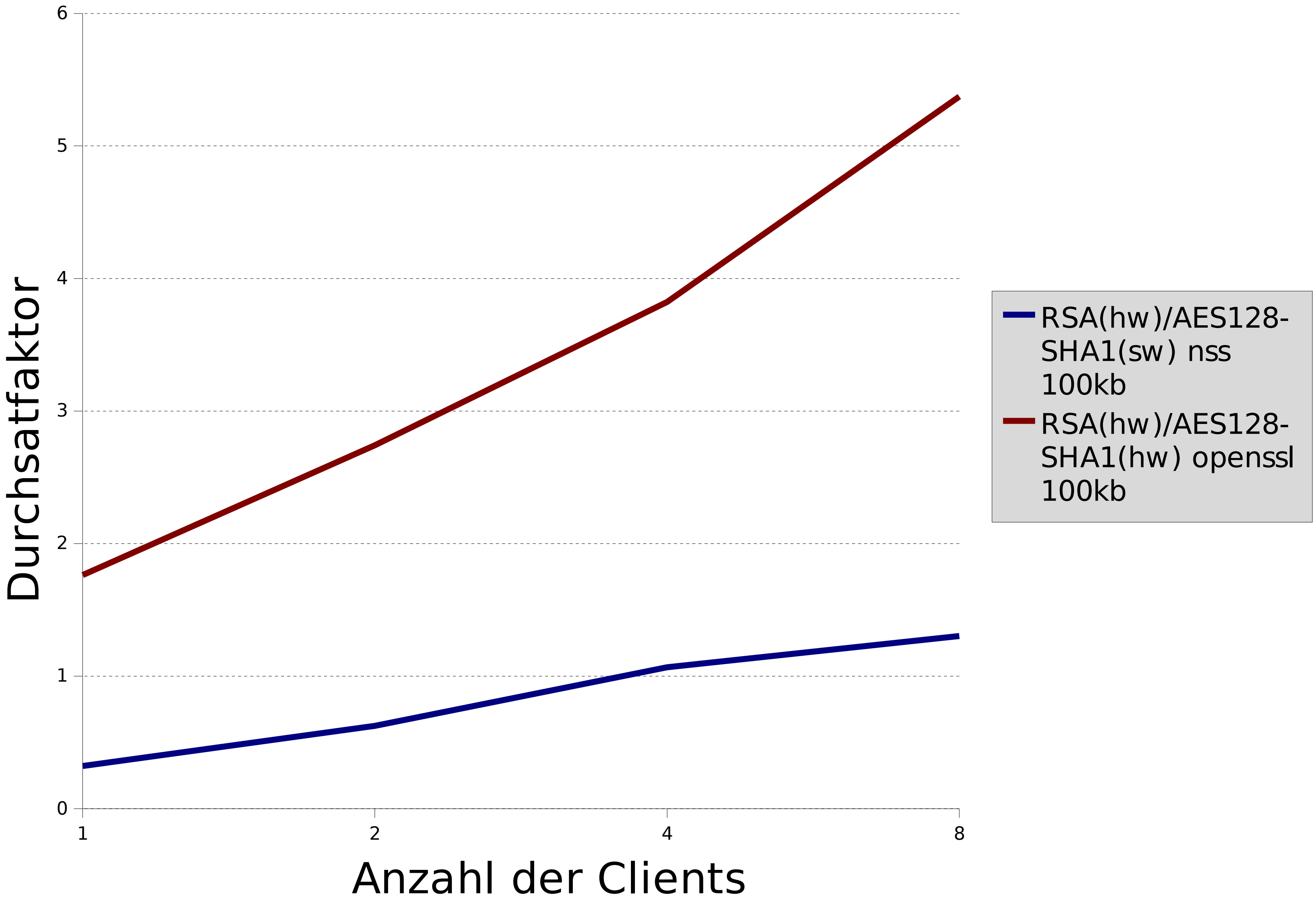
Note: These are not official performance data

Apache mit mod_nss ↔ Apache mit mod_ssl für RSA/RC4/MD5



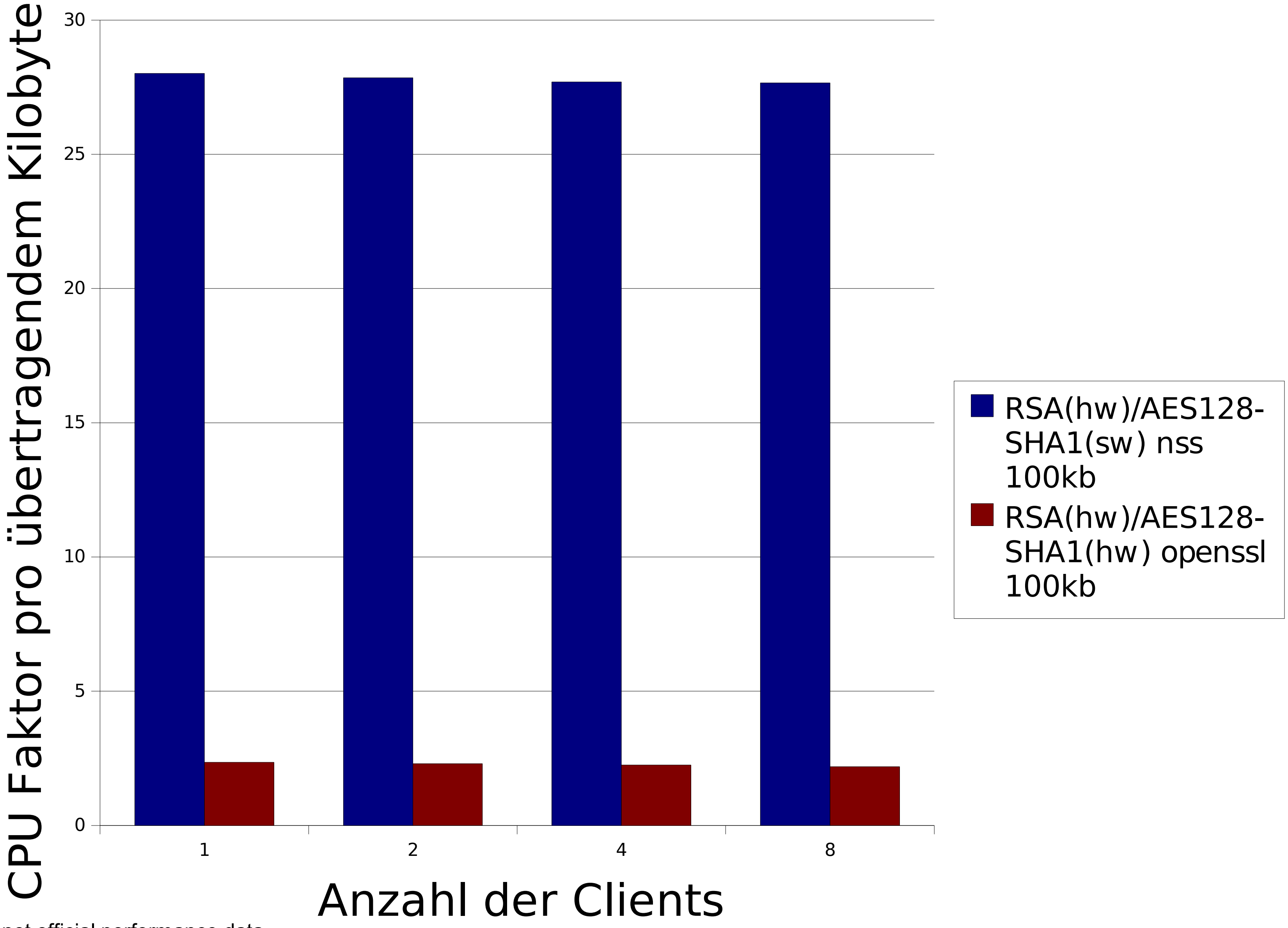
Note: These are not official performance data

Apache mit mod_nss ↔ Apache mit mod_ssl für RSA/AES-128/SHA



Note: These are not official performance data

Apache mit mod_nss ↔ Apache mit mod_ssl für RSA/AES-128/SHA



Note: These are not official performance data

First Comparison of Apache on Linux for System z: NSS vs. OpenSSL . . .

Zusammenfassung:

NSS:HW Unterstützung von System z (CEX2 und CPACF) nutzbar.

Apache mit NSS funktioniert.

Apache mit NSS kann RSA HW Unterstützung nutzen.

Apache mit NSS nutzt derzeit nicht CPACF (bis auf Ausnahme TDES beim Initialisieren des Apaches).

Im Moment keine Empfehlung Apache mit NSS auf Linux for System z einzusetzen: In unserer Testumgebung ist in allen getesteten Szenarien OpenSSL „besser“ (bzgl. Durchsatz und Kosten) als NSS (für alle getesteten CipherSuites, mit oder ohne Hardware Unterstützung).

Grund für schlechteres Abschneiden von mod_nss-NSS-openCryptoki ist noch unklar (mod_nss, NSS, openCryptoki?). (SW: Implementierung od. Pfadlänge? Nichtnutzung der CPACF: Design von mod_nss?)

Es ist noch Arbeit zu investieren - optimistisch.

Agenda

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- Setup for cryptographic hardware support on z10
- Access of cryptographic hardware support with Linux for System z
- In-kernel cryptographical support
- Linux applications using HW cryptographic Support
- New tool
- Cryptographic support with Java on Linux for System z
- NSS Network Security Services
- **openssh**
- Secure key cryptography

openssh and Cryptographic Hardware Support

- openssh is a way to provide a secure login to a remote server.
- openssh uses RSA, DES, Tripple DES, AES, …
- openssh uses OpenSSL
- Today (openssh version 4.2 in SUSE SLES 10 SP1 does not use dynamic engine loading support of OpenSSL and does not provide a way to explicitly specify the engine ibmca.
  (-> all encryption is done in software w/o CEX2x or CPACF)
- Starting with opnessh version 4.4 there is a flag --with-ssl-engine for the configure step to benefit from OpenSSL dynamic engine support.
  - If distributors will build openssh with this new flag then available hardware support with System z is automatically used.
  - Check and Information: tbd soon

Agenda

- Clear key versus secure key (general)
- Hardware support for cryptographic operations on System z
- Setup for cryptographic hardware support on z10
- Access of cryptographic hardware support with Linux for System z
- In-kernel cryptographical support
- Linux applications using HW cryptographic Support
- New tool
- Cryptographic support with Java on Linux for System z
- NSS Network Security Services
- openssh
- **Secure key cryptography**

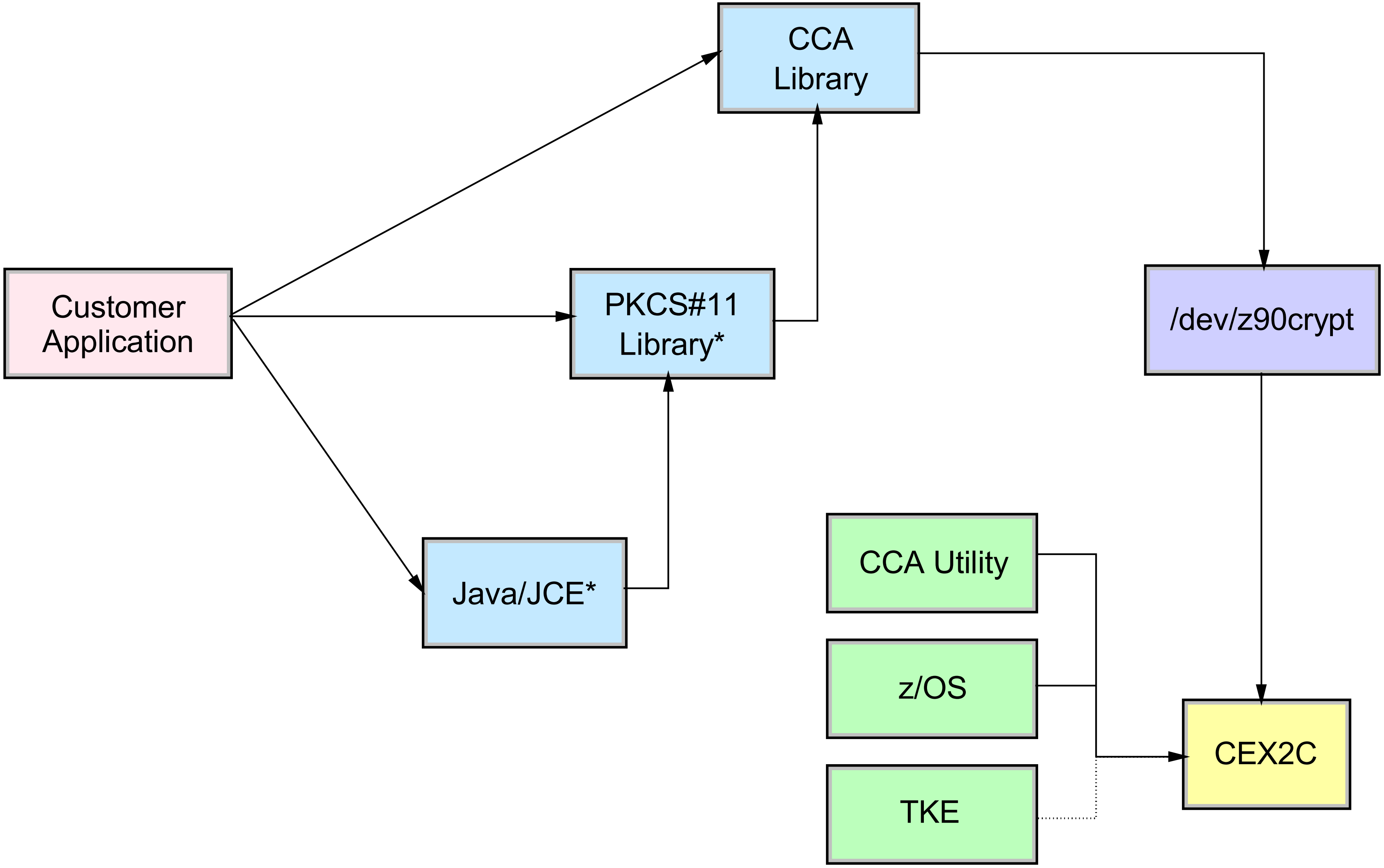
Secure key cryptography

Since 2007: Support for secure key cryptography for Linux for System z
Linux can benefit from capabilities of Crypto Express2

Solution consists of:
- Crypto Express2 configured as CEX2C
- Device driver z90crypt
- Common Cryptographic Architecture (CCA) libraries
- Only for 64 bit linux

Management of crypto keys and crypto hardware:
- Using z/OS ICSF
- Using a Trusted Key Entry (TKE) console with connection to a z/OS
- Using a new Linux CCA utility
- Using a Trusted Key Entry (TKE) console with connection to a new Linux
- CCA utility

Secure key cryptographic solution



Secure key cryptography: installation

- You need all software as mentioned for clear key cryptography
- Ensure, that CCA libraries are installed

```
gnirss@tmcc-123-168:~> rpm -qa | grep xcrypto
xcryptolinzGA-3.28-rc08
```

- Package xcryptolinzGA-3.28-rc08.s390x.rpm is available from

  http://www.ibm.com/security/cryptocards/pcixcc/ordersoftware.shtml

- Package contains a README.linz file with all relevant information (installation notes, description or syntax, as well as usage notes)
- Content of package:
  - CCA libraries
  - Installation verification program (ivp.e)
  - TKE Catcher (TKEC)
    responds to commands fro a remote TKE workstation
  - Panel CLI (panel.exe)
    is a command line utility to manage keys

Secure key: TKE catcher

- The TKE catcher is a program running on Linux for System z that allows remote access from the workstation to administrate crypto cards and the according keys.
- To make use of the TKE catcher, the TKE must be enabled to access the system via s390 SE panel and using port 50003.
- Control Domain Index and TKE commands must be permitted for the used crypto adapters.
- Consider the following 3 cases for using TKE for Linux for System z:
  - Environment with Linux and z/OS LPARs sharing a Crypto Express2 adapters
    - Difficult environment if you intend to use TKE catcher to administrate the crypto queues accessible by Linux and the z/OS TKE for the crypto queues accessible by z/OS. TKE catcher can not figure out whether there is a z/OS LPAR and whether crypto is being configured with z/OS TKE .
    - To avoid conflicts, we recommend to use the z/OS TKE in such an environment.
  - Environment with Linux and z/OS LPARs with each exclusive use of Crypto Express2 adapters
    - Usage of TKE catcher is possible.
    - Note: Situation gets dificult if environment is reconfigured to share adapters.
  - Linux for System z exclusive environment
    - Using TKE with TKE catcher is most secure way to administrate crypto infrastructure.

Zusammenfassung

Kryptographie mit Linux for System z ist interessant . . .

Danke für Ihre Aufmerksamkeit

Fragen ?

Quellen/Literatur

[1] Wikipedia

[2] RedBook: Security on z/VM – SG24-7471

[3] Stefan Kirchner: Integration spezifischer Network Security Services unter Linux auf System z im Vergleich mit OpenSSL – to be published

[4] IBM Techdocs: First Comparison of Apache with NSS vs Apache with OpenSSL on Linux for System z Techdocs – to be published.