GSE GUIDE SHARE EUROPE
Deutsche Region

GS10
Linux on z Systems – News

Hamburg, 24.10.2017

Dr. Manfred Gnirss
Arwed Tschoeke

Z ATS

IBM Client Center, Boeblingen, Germany

11th European GSE/IBM Technical University for z/VSE, z/VM, KVM and Linux on IBM z Systems

23 – 25 October 2017 in Hamburg

Le Méridien Hotel
Hamburg

© PG / GSE

Trademarks & Disclaimer

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries. For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

IBM, the IBM logo, BladeCenter, Calibrated Vectored Cooling, ClusterProven, Cool Blue, POWER, PowerExecutive, Predictive Failure Analysis, ServerProven, System p, System Storage, System x , z Systems, WebSphere, DB2 and Tivoli are trademarks of IBM Corporation in the United States and/or other countries. For a list of additional IBM trademarks, please see http://ibm.com/legal/copytrade.shtml.

The following are trademarks or registered trademarks of other companies: Java and all Java based trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries or both Microsoft, Windows,Windows NT and the Windows logo are registered 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 or both. Linux is a trademark of Linus Torvalds in the United States, other countries, or both. Cell Broadband Engine is a trademark of Sony Computer Entertainment Inc. InfiniBand is a trademark of the InfiniBand Trade Association. Other company, product, or service names may be trademarks or service marks of others.

NOTES: Linux penguin image courtesy of Larry Ewing (lewing@isc.tamu.edu) and The GIMP

Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Users of this document should verify the applicable data for their specific environment. IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply. Information is provided “AS IS” without warranty of any kind. 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.

Trademarks & Disclaimer

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 are suggested US list prices and are subject to change without notice. Starting price may not include a hard drive, operating system or other features. Contact your IBM representative or Business Partner for the most current pricing in your geography. Any proposed use of claims in this presentation outside of the United States must be reviewed by local IBM country counsel prior to such use. The information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any

Notice Regarding Specialty Engines

Any information contained in this document regarding Specialty Engines (“SEs”) and SE eligible workloads provides only general descriptions of the types and portions of workloads that are eligible for execution on Specialty Engines (e.g., zIIPs, zAAPs, and IFLs). IBM authorizes customers to use IBM SE only to execute the processing of Eligible Workloads of specific Programs expressly authorized by IBM as specified in the “Authorized Use Table for IBM Machines” provided at www.ibm.com/systems/support/machine_warranties/machine_code/aut.html(“AUT”).
No other workload processing is authorized for execution on an SE.
IBM offers SEs at a lower price than General Processors/Central Processors because customers are authorized to use SEs only to process certain types and/or amounts of workloads as specified by IBM in the AUT

Acknowledgement

Our very best thanks belong to

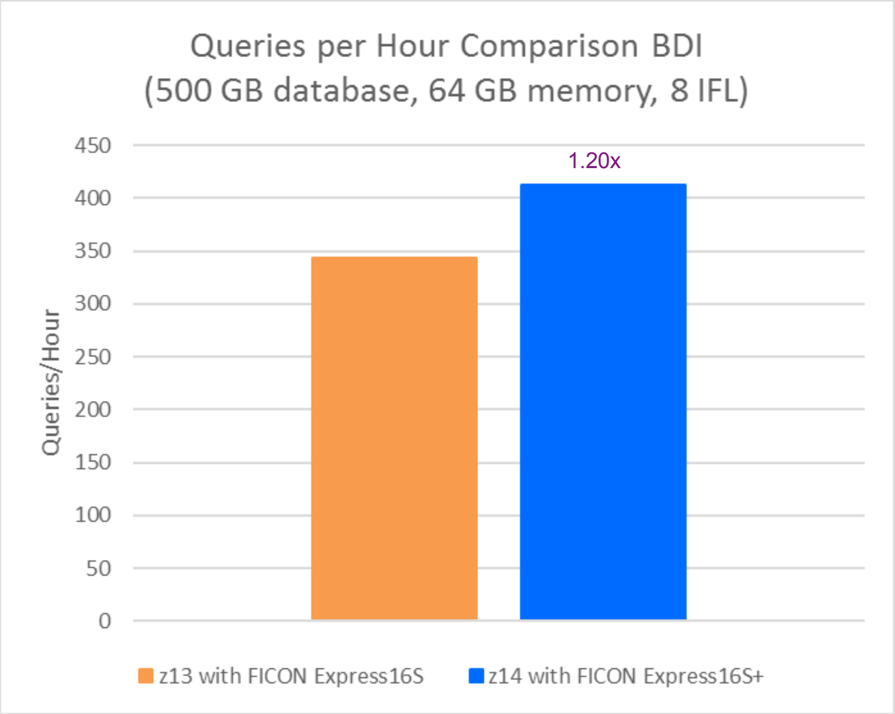
Martin Schwidefsky

and all the others

who contributed to this session

DB2 LUW 11.1.1 Performance with FICON Express16S+ Cards

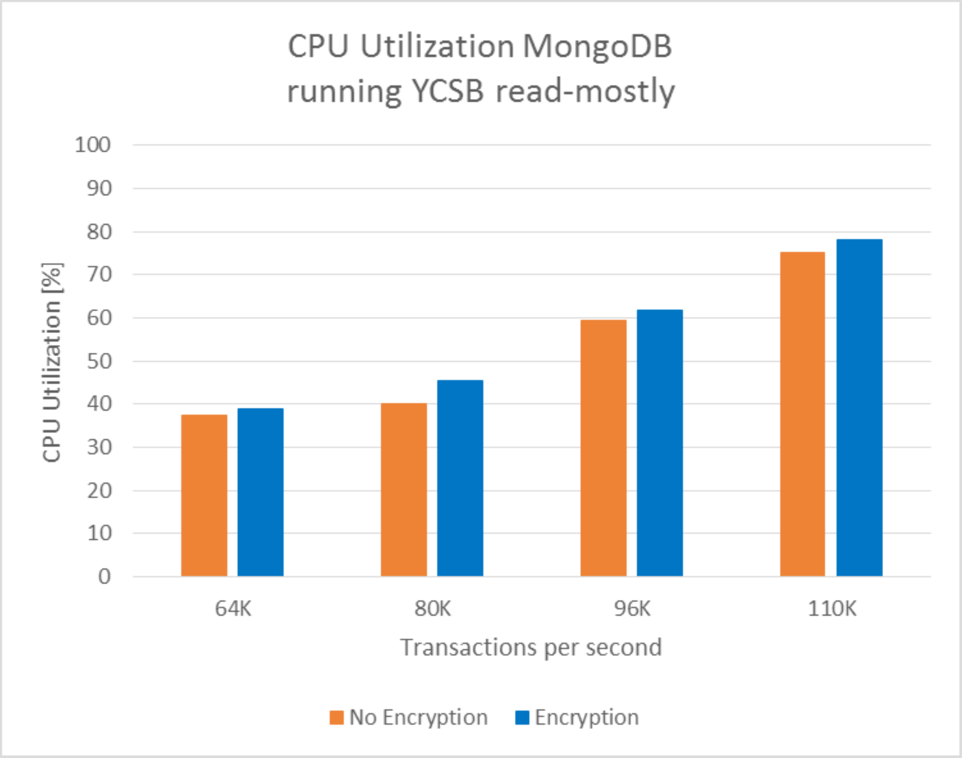
Run the BDI benchmark on DB2 LUW 11.1.1 with up to 20% more throughput using FICON Express16S+ cards on z14 compared to using FICON Express16S cards on z13



Disclaimer: Performance results based on IBM internal tests running the BDI benchmark, which is based on TPC-DS, on DB2 LUW with BLU Acceleration. The BDI benchmark was configured to run a fixed sequence of queries. DB2 database size was 500 GB. Results may vary. z13 configuration: LPAR with 8 dedicated IFLs, 64GB memory, and 11 TB LUN on IBM FlashSystem 900 attached via FICON Express16S cards, RHEL 7.3 (SMT mode) running DB2 LUW 11.1.1, IBM Java 1.8, and BDI. z14 configuration: LPAR with 8 dedicated IFLs, 64GB memory, and 11 TB LUN on IBM FlashSystem 900 attached via FICON Express16S+ cards, RHEL 7.3 (SMT mode) running DB2 LUW 11.1.1, IBM Java 1.8, and BDI.

CPU Overhead with Pervasive Encryption for MongoDB on z14

Run the read-mostly workload of the YCSB benchmark on MongoDB Enterprise Edition 3.4.1 with only 6% CPU overhead on average when enabling pervasive encryption on a z14 LPAR



<u>Disclaimer</u>: Performance result is extrapolated from IBM internal tests running MongoDB Enterprise Edition 3.4.1 with and without SSL and database encryption driven remotely by 512 total threads of Yahoo! Cloud Serving Benchmark (YCSB) using the workload read-mostly (95% read, 5% update) and a record size of 5KB. Two external x86 blade-servers, each with 4 independent YCSB instances stressed the MongoDB database simultaneously. YCSB was configured to generate constant throughput rates. RSA 4096 bit key for SSL configuration of MongoDB. GCM based ciphers were used for SSL. Database stored via dm-crypt using aes-xts-plain64. CPU utilization for the pervasive encryption case was projected by scaling the achieved throughput with pervasive encryption to the throughput achieved without encryption. Results may vary. z14 configuration: LPAR with 8 dedicated IFLs, 256GB memory, 40 GB DASD storage, RHEL 7.3 (SMT mode), OpenSSL 1.0.1e-fips, 50GB database on IBM DS8000 storage.

Z Data Compression (zEDC)

Customer advantages

- If the workload fits, it rocks!
- Sweet spot workloads
  - **Large request sizes** to compensate for request latency to PCI adapter
  - Products using **gzip/deflate standard compression** which is what zEDC accelerates
- Sweet spot workload examples
  - Database backup & restore
  - IBM Java 7.1 and Java 8
  - IBM MQ
  - IBM WebSphere Application Server
  - IBM DB2
  - Apache Kafka *and more*

OS requirements for zEDC exploitation

(FYI, z14 XXX exploitation requirements of Linux distros are different, see p19)

- SLES 12 SP3 and later
- RHEL 7.3 and later
- Ubuntu 16.04.03 and later
- z/VM V6.4

Proof Points: Z Data Compression (zEDC)

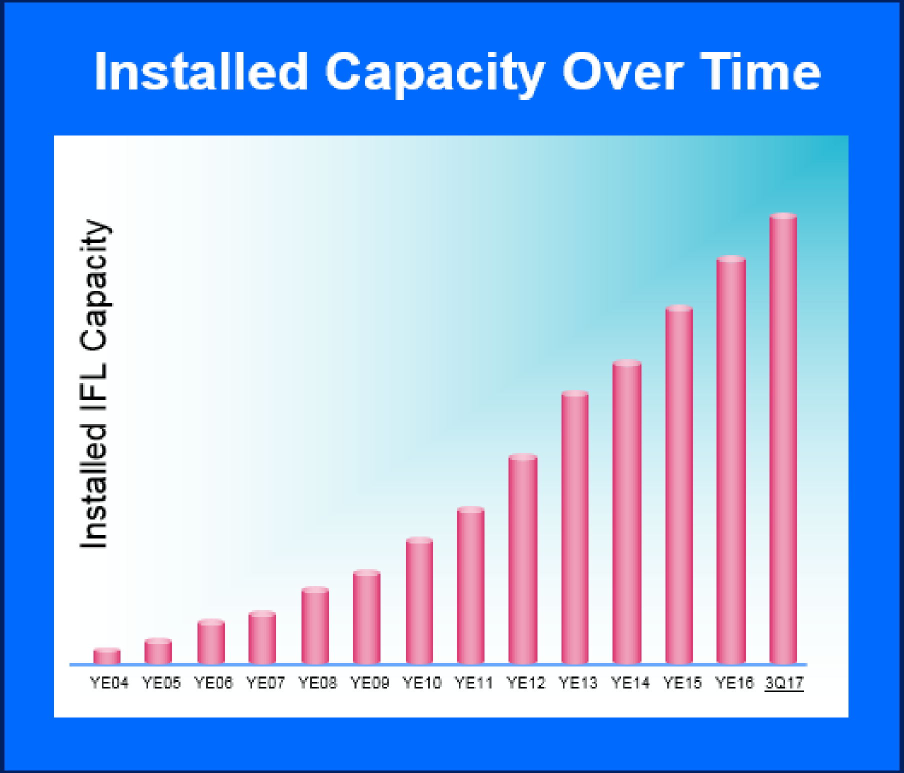
Database Backup & Restore (z14 vs. x86)

- Operators can perform database backup up to **11.9x** faster and database restore up to 2.4x faster for **DB2 LUW** 11.1.1 on a z14 LPAR using zEDC Express versus a compared x86 platform using software compression.
- Operators can perform database backup with up to **75%** lesser CPU utilization and database restore with up to 71% lesser CPU utilization for **DB2 LUW** 11.1.1 on a z14 LPAR using zEDC Express versus a compared x86 platform using software compression.
- Operators can perform database dump up to **3.5x** faster and database restore up to 1.1x faster for **MongoDB** Enterprise Edition 3.4.6 on a z14 LPAR using zEDC Express versus a compared x86 platform using software compression.
- Operators can perform database dump with up to **84%** lesser CPU utilization and database restore with up to 9% lesser CPU utilization for **MongoDB** Enterprise Edition 3.4.6 on a z14 LPAR using zEDC Express versus a compared x86 platform using software compression.

Linux on IBM z Systems in 3Q2017

*Installed Linux MIPS at 40% CAGR**

- 29.5% of Total installed MIPS run Linux as of 3Q17
- Installed IFL MIPS increased by 19% YTY from 3Q16 to 3Q17
- 50% of IBM Z Enterprises have IFL’s installed as of 3Q17
- 91 of the top 100 IBM Z Enterprises are running Linux on z as of 3Q17 **
- 37% of all IBM Z servers have IFLs
- 59% of new FIE/FIC IBM Z Accounts run Linux



Installed Capacity Over Time

* *Based on YE 2003 to YE 2016*   ***Top 100 is based on total installed MIPS*

Linux on z Systems distributions

What is available today

Linux on z Systems distributions

- **SUSE Linux Enterprise Server 10**
  - 07/2006 SLES10 GA: Kernel 2.6.16, GCC 4.1.0
  - 04/2011 SLES10 SP4; **EOS 31 Jul. 2013; LTSS: 30 Jul. 2016**
- **SUSE Linux Enterprise Server 11**
  - 03/2009 SLES11 GA: Kernel 2.6.27, GCC 4.3.3
  - 07/2015 SLES11 SP4: Kernel 3.0, GCC 4.3.4; EOS 31 Mar. 2019; LTSS: 31 Mar. 2022
- **SUSE Linux Enterprise Server 12**
  - 10/2014 SLES12 GA: Kernel 3.12, GCC 4.8
  - 11/2016 SLES12 SP2: Kernel 4.4, GCC 4.8
  - Last SP: EOS 31 Oct. 2024; LTSS: 31 Oct. 2027

Linux on z Systems distributions

- **Red Hat Enterprise Linux AS 5**
  - 03/2007 RHEL5 GA: Kernel 2.6.18, GCC 4.1.0
  - 09/2014 RHEL5 Update 11; EOS 31 Mar. 2017; ELS: 30 Nov. 2020
- **Red Hat Enterprise Linux AS 6**
  - 11/2010 RHEL6 GA: Kernel 2.6.32, GCC 4.4.0
  - 03/2017 RHEL6 Update 9; EOS 30 Nov. 2020; ELS: tbd
- **Red Hat Enterprise Linux AS 7**
  - 06/2014 RHEL7 GA: Kernel 3.10, GCC 4.8
  - 08/2017 RHEL7 Update 4; EOS 30 Jun. 2024; ELS: tbd

Linux on z Systems distributions

- **Ubuntu 16.04 (Xenial Xerus)**
  - Canonical and IBM announced an Ubuntu based distribution on LinuxCon 2015 in Seattle
  - 04/2016 Ubuntu 16.04 GA: Kernel 4.4, GCC 5.3.0+ LTS-Release
  - 10/2016 Ubuntu 16.10 GA: Kernel 4.8, GCC 6.2.0+, EOS 04/2017
  - 04/2017 Ubuntu 17.04 GA: Kernel 4.10, GCC 6.3.0+
  - Lifecycle:
    - Regular releases every 6 months and supported for 9 months
    - LTS releases every 2 years and supported for 5 years
    - LTS enablement stack will provide newer kernels within LTS releases
    - http://www.ubuntu.com/info/release-end-of-life

Linux distros toleration plans and exploitation plans for IBM z14 XXX

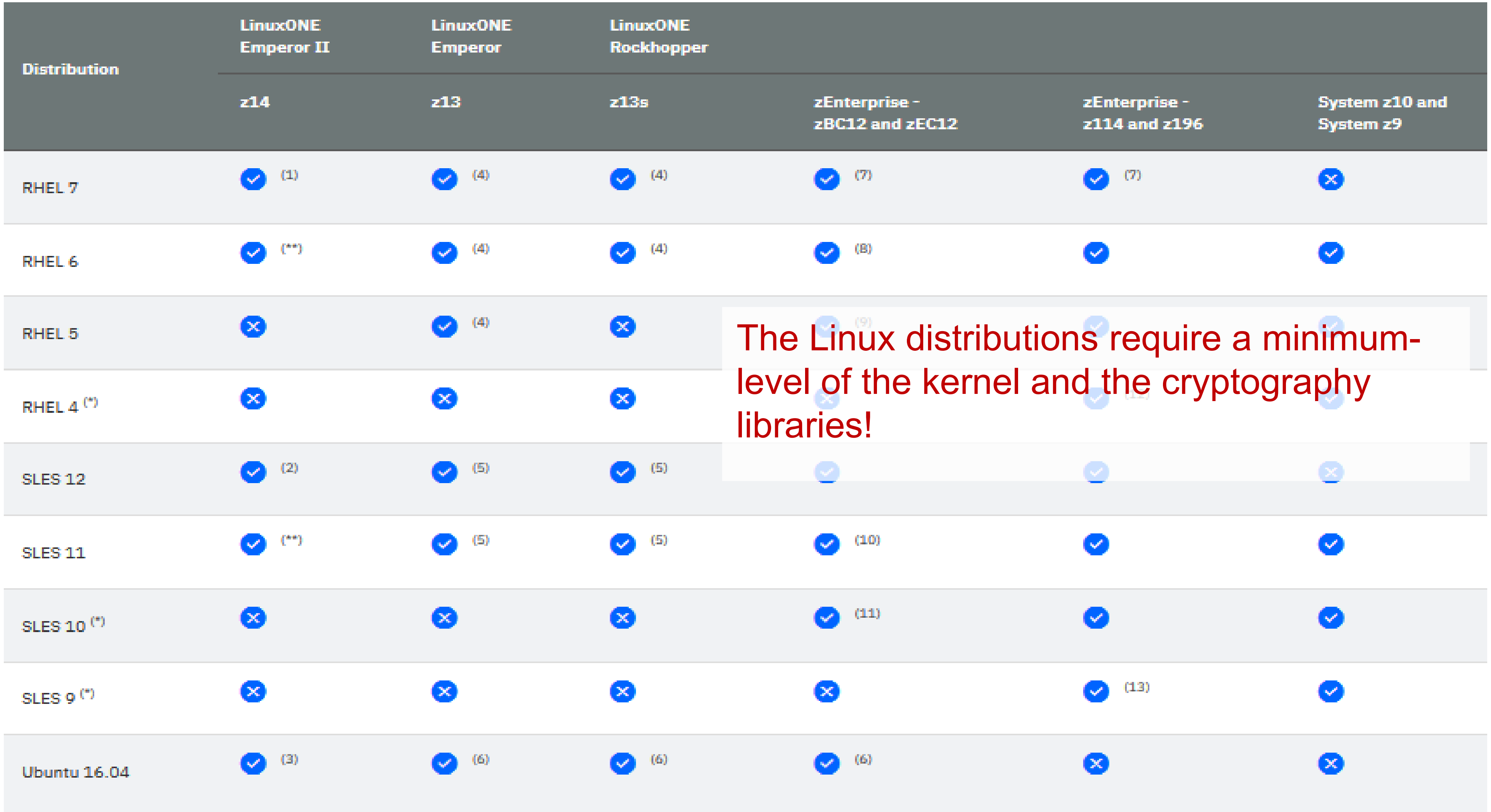
Canonical	▪ Toleration[1] available with Ubuntu 16.04 LTS – Multithreading with next generation SMT, FICON Express16S+, 8 TB (z/VM up to 2 TB) ▪ Toleration[1] expected with KVM in Ubuntu 16.04 LTS ▪ Exploitation[1,2] to be done – SIMD, CPACF and CryptoExpress6S performance, GCM encryption, protected key encryption, support of true random number generator, pause-less garbage collection
Red Hat	▪ Toleration[1] available with RHEL 7.3 with service and expected with RHEL 6.9[3] with service update – Multithreading with next generation SMT, FICON Express16S+, 8 TB (z/VM up to 2 TB) ▪ Exploitation[1,2] to be done – SIMD, CPACF and CryptoExpress6S performance, GCM encryption, protected key encryption, support of true random number generator, pause-less garbage collection
SUSE	▪ Toleration[1,2] available with SUSE SLES 12 SP2 with service and expected with SUSE SLES 11 SP4[3] with service update – Multithreading with next generation SMT, FICON Express16S+, 8 TB (z/VM up to 2 TB) ▪ Toleration[1] expected with KVM in SLES 12 SP2 with service ▪ Exploitation[1,2] to be done – SIMD, CPACF and CryptoExpress6S performance, GCM encryption, protected key encryption, support of true random number generator, pause-less garbage collection

[1] *For minimum required and recommended distribution levels see the IBM Z – "Tested platforms" website.*

[2] *IBM is working with the Linux partner to support selected levels of the distribution on z14.*

[3] *SMT is not supported with this Linux distribution, SMT is supported only via z/VM.*

Tested platforms – supported Linux distributions



Distribution	LinuxONE Emperor II	LinuxONE Emperor	LinuxONE Rockhopper			
	z14	z13	z13s	zEnterprise - zBC12 and zEC12	zEnterprise - z114 and z196	System z10 and System z9
RHEL 7	✓ (1)	✓ (4)	✓ (4)	✓ (7)	✓ (7)	✗
RHEL 6	✓ (**)	✓ (4)	✓ (4)	✓ (8)	✓	✓
RHEL 5	✗	✓ (4)	✗	✓ (9)	✓	✓
RHEL 4 (*)	✗	✗	✗	✗	✓ (12)	✓
SLES 12	✓ (2)	✓ (5)	✓ (5)	✓	✓	✗
SLES 11	✓ (**)	✓ (5)	✓ (5)	✓ (10)	✓	✓
SLES 10 (*)	✗	✗	✗	✓ (11)	✓	✓
SLES 9 (*)	✗	✗	✗	✗	✓ (13)	✓
Ubuntu 16.04	✓ (3)	✓ (6)	✓ (6)	✓ (6)	✗	✗

The Linux distributions require a minimum-level of the kernel and the cryptography libraries!

(**)IBM is working with the Linux partner to support selected levels of the distribution on z14 and z14 XXX.

- *RHEL6 support is planned to be based on a service update of RHEL 6.9*
- *SLES 11 support is planned to be based on a service update of SLES11 SP4*

Note: the required patch levels and additional details will be provided soon.

View webpage for additional footnotes and details.

The information is regularly updated, see actual information at [Tested Platforms]

Linux on z Systems distributions

- Please check the tested-platforms web link for minimum required kernel levels
- Notes about IBM z14
  - RHEL5 is **not** supported on IBM z14, SLES10 has not been supported on IBM z13 already
  - Tested platforms current has the following footnote on z14:
“IBM is working with the Linux partner to support selected levels of the distribution on z14; details will be provided soon.”
  - RHEL6 and SLES11 required at least one small patch
  - RHEL7 and SLES12 run fine on z14, but do not have “z14” in the ELF platform name

Current Linux on z Systems Technology

Key features & functionality already contained in the SuSE, Red Hat and Ubuntu Distributions

Tag legend

- Supported distributions

  x.y for SUSE SLES <X> Service Pack <Y>, e.g. 12.1 for SLES12 SP1

  x.y for RHEL <x> Update <y>, e.g. 7.2 for RHEL7.2

  x.y for Ubuntu x.y, e.g. 16.04 for Ubuntu 16.04 LTS

- Suppored environments

  LPAR usable for systems running under LPAR

  z/VM usable for guests running under z/VM

  KVM usable for guests running under KVM

IBM z13 Support

- **Vector extension facility (kernel 3.18)** LPAR z/VM KVM 12.1 7.2 16.04
  - Also known as single-instruction, multiple data (**SIMD**)
  - 32 128-bit vector registers are added to the CPU
  - 139 new instructions to operate on the vector registers
  - User space programs can use vectors to speed up all kinds of functions, e.g. string functions, crc checksums, …
- **CPU multi threading support (> kernel 3.19)** LPAR 12.1 7.2 16.04
  - Also known as simultaneous multi-threading (**SMT**)
  - Once enabled the multi threading facility provides multiple CPUs for a single core.
  - The CPUs of a core share certain hardware resource such as execution units or caches
  - Avoid idle hardware resources, e.g. while waiting for memory

IBM z13 Support

- **Extended number of AP domains (kernel 3.18)** LPAR z/VM 12.1 7.2 16.04
  - AP crypto domains in the range 0-255 will be detected
- **Crypto Express 5S cards (kernel 4.0)** LPAR z/VM 12.1 7.2 16.04
  - New generation of crypto adapters with improved performance
- **z13 cache aliasing (kernel 4.0)** 12.1 7.2 16.04
  - Shared objects mapped to user space need to be aligned to 512KB for optimum performance on z13
- **Drawer scheduling domain level (kernel 4.8)** LPAR 7.4 16.10
  - Add another scheduling domain to reflect the exact machine structure for z13.
  - There are now: node, drawer, book, MC and SMT domains
  - Older kernel versions folded drawer and nodes into books

Compiler Toolchain

- **zEC12/zBC12 exploitation CPU (gcc 4.8)** LPAR z/VM KVM 12.0 7.0 16.04
  - Use option -march=zEC12 to utilize the instructions added with zEC12
  - Use option -mtune=zEC12 to schedule the instructions appropriate for the pipeline of zEC12
  - Transactional memory support, Improved branch instructions
- **z13/z13s exploitation CPU (gcc 5.2)** LPAR z/VM KVM 12.1 7.3 16.04
  - Use option -march=z13 to utilize the instructions added with z13
  - Use option -mtune=z13 to schedule the instructions appropriate for the pipeline of z13
  - SLES12SP1 support with the gcc 5.3.1 toolchain module
- **z14 exploitation CPU (gcc 7.1)** LPAR z/VM KVM
  - Use option -march=z14 to utilize the instructions added with z14
  - Use option -mtune=z14 to schedule the instructions appropriate for the pipeline of z14

Miscellaneous new kernel features

- **Support for IPL Device in Any Sub-Channel Set (k. 4.4)** LPAR z/VM KVM 7.4 12.2 16.04
  - Allows to boot the OS from a device with an address '0.x.yyyy' with x != 0
- **Add a statistic for diagnose calls (kernel 4.4)** LPAR z/VM KVM 12.2 16.04
  - Provide the number of diagnose calls per CPU via '/sys/kernel/debug/diag_stat'
  - Useful to find congestion problems, watch the values for diag 044 and diag 09c
  - The high value on CPU #0 is due to a timing loop at IPL

```
# cat /sys/kernel/debug/diag_stat
                CPU0       CPU1       CPU2       CPU3
diag 008:          0          0          0          0    Console Function
diag 00c:          0          0          0          0    Pseudo Timer
diag 010:          0          0          0          0    Release Pages
diag 014:          0          0          0          0    Spool File Services
diag 044:     663700          1          1          1    Voluntary Timeslice End
diag 064:          0          0          0          0    NSS Manipulation
diag 09c:          3          2          3          1    Relinquish Timeslice
diag 0dc:          0          0          0          0    Appldata Control
...
```

Miscellaneous new kernel features

- **LPAR offset handling (kernel 4.8)** LPAR 7.4 12.2
  - Initialize the Linux system clock with the physical TOD clock, effectively removing the LPAR offset
  - Get Linux to a consistent time base in regard to other machines
- **2GB pages for hugetlbfs (kernel 4.8)** LPAR KVM 7.4 16.10
  - Extend the huge page support to allow 2GB huge pages next to 1MB large pages
  - Access 2GB pages either through the mmap() or SysV shared memory system calls
  - Transparent huge pages are not affected by this, they stay at 1MB pages
  - Promises to speed up Java with large heap sizes and databases with big SGAs
- **Vector optimization for CRC32 (kernel 4.8)** LPAR z/VM KVM 16.10
  - Cyclic redundancy checks (CRCs) are error-detecting codes commonly used in network protocols and file systems
  - Speed up the in-kernel CRC32 code by by use of vector instructions

Vector optimization for CRC32 in the kernel

LPAR z/VM 16.10

- **Inner loop of crc32_be:**

```
# %v9 contains a magic constant, %v1-%v4 the intermediate checksum
LOOP:   VLM     %v5,%v8,0,%r3      # load next 64 bytes
        VGFMAG  %v1,%v9,%v1,%v5        # 1st GF(2) multiplication
        VGFMAG  %v2,%v9,%v2,%v6        # 2nd    GF(2) multiplication
        VGFMAG  %v3,%v9,%v3,%v7        # 3rd    GF(2) multiplication
        VGFMAG  %v4,%v9,%v4,%v8        # 4th    GF(2) multiplication
        aghi        %r3,64             # buf = buf + 64
        aghi        %r4,-64            # len = len - 64
        cghi        %r4,64             # check remaining length
        jnl      LOOP                  # loop if >= 64 bytes remain
```

- 9 instructions to do crc32 for 64 bytes

Kernel features – DASD improvements

- **Query host access to volume (kernel v4.7)** LPAR z/VM 7.4 12.2
  - Add an interface to query if a DASD volume is online to another operation system instance.
- **DASD quick format mode for use with dasdfmt (kernel v4.7)** LPAR z/VM 7.4
  - Add an option to re-initialize an already formatted DASD device, just write VTOC and the label
- **DASD channel path aware error recovery (kernel v4.10)** LPAR z/VM 7.4 17.04
  - Improve robustness of the DASD device driver with multiple channel paths
  - A channel patch with repeated Interface-control-checks (IFCC), channel-control-checks (CCCs), or loss of high-performance-FICON (HPF) will be removed as long as other paths are available

Kernel features: PCI improvements

LPAR z/VM 16.10

- **PCI call logical-processor query interface (kernel v4.6)**
  - Provide a user space interface to submit query requests for installed PCI functions.

LPAR z/VM 7.4 16.10

- **PCI function-type specific measurement data (kernel v4.7)**
  - Enhances the statistics interface to display PCI function-specific measurement data for IBM z13 and later

LPAR z/VM 7.4 17.04

- **PCI unique UIDs for domain enumeration (kernel v4.10)**
  - Use the PCI UID for the domain field of the PCI bus-id if firmware guarantees the uniqueness of these values

Kernel features – crypto support

- **zcrypt workload balancing (kernel 4.10)** LPAR z/VM 17.04
  - The complexity of a cryptographic request determines how long it will take
  - Add requests weights and adapter speed ratings to find a better balance for the work between cards
- **zcrypt multi-domain support (kernel 4.10)** LPAR 17.04
  - The AP bus infrastructure used to support only one cryptographic domain, the associated queue of the card for the one domain has been equivalent to the card
  - Add code to differ between a card and the queues of a card, to allow the use of multiple cryptographic domains simultaneously
  - sysfs interface stays compatible to the old layout
  - Existing user space code continues to work with the default domain

Container Support for Docker

LPAR	12.1
z/VM	7.3
KVM	16.04

- **Docker provides lightweight containers**
  - Self contained set of files to package an application with all of its dependencies
- **Applications in containers share the OS kernel**
  - **No virtualization – no virtualization overhead**
- **“Build, Ship, and Run Any App, Anywhere”**
  - One implementation of a container solution
  - Maintained by Docker, Inc.
  - Docker Hub cloud-based registry service, see https://hub.docker.com
- **Power tool to build, modify, deploy, run, manage containers**
  - E.g. “docker run hello-world”

docker

Future Linux on z Systems Technology

Software which has already been developed
and integrated into the upstream packages
- but is **not yet available** in any
Enterprise Linux Distribution

SMC-R and RoCE

LPAR z/VM

- Shared Memory Communications over RDMA (SMC-R) is a protocol that allows applications to exploit RDMA (RoCE) with the socket interface
- A first version of the Linux code is now upstream with kernel 4.11-rc1
  - The Linux variant is currently **incompatible** with the z/OS version
  - More work on both the Linux and the z/OS side is required to connect Linux to z/OS via SMC-R

SMC-R and RoCE

LPAR z/VM

- The Linux support for SMC-R uses a new address family **AF_SMC**
  - The addressing scheme is the same as TCP, to “port” an application to SMC-R simply replace AF_INET with AF_SMC:

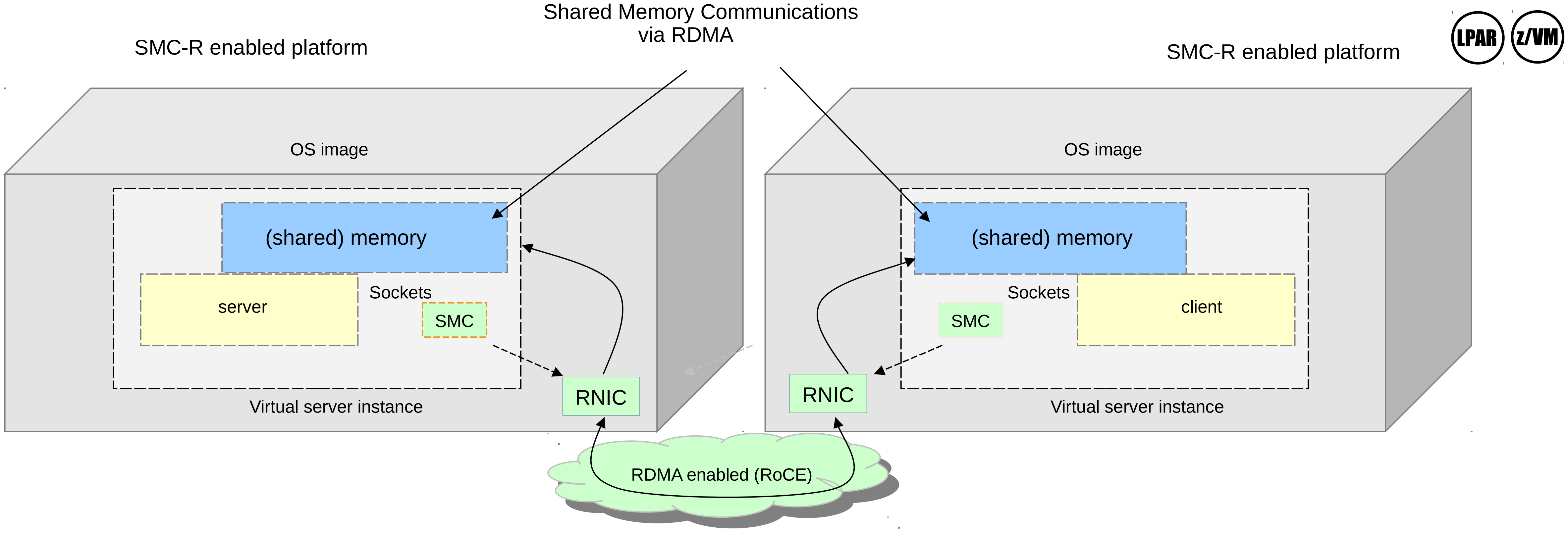
    ```
    tcp_socket = socket(AF_INET, SOCK_STREAM, 0);
    ```

    by

    ```
    tcp_socket = socket(AF_SMC, SOCK_STREAM, 0);
    ```

  - Alternatively a preload library available in package SMC Tools at https://ibm.biz/BdiZ5m can be used to intercept the socket call
  - Automatic fallback to AF_TCP if the connection could not be established via SMC

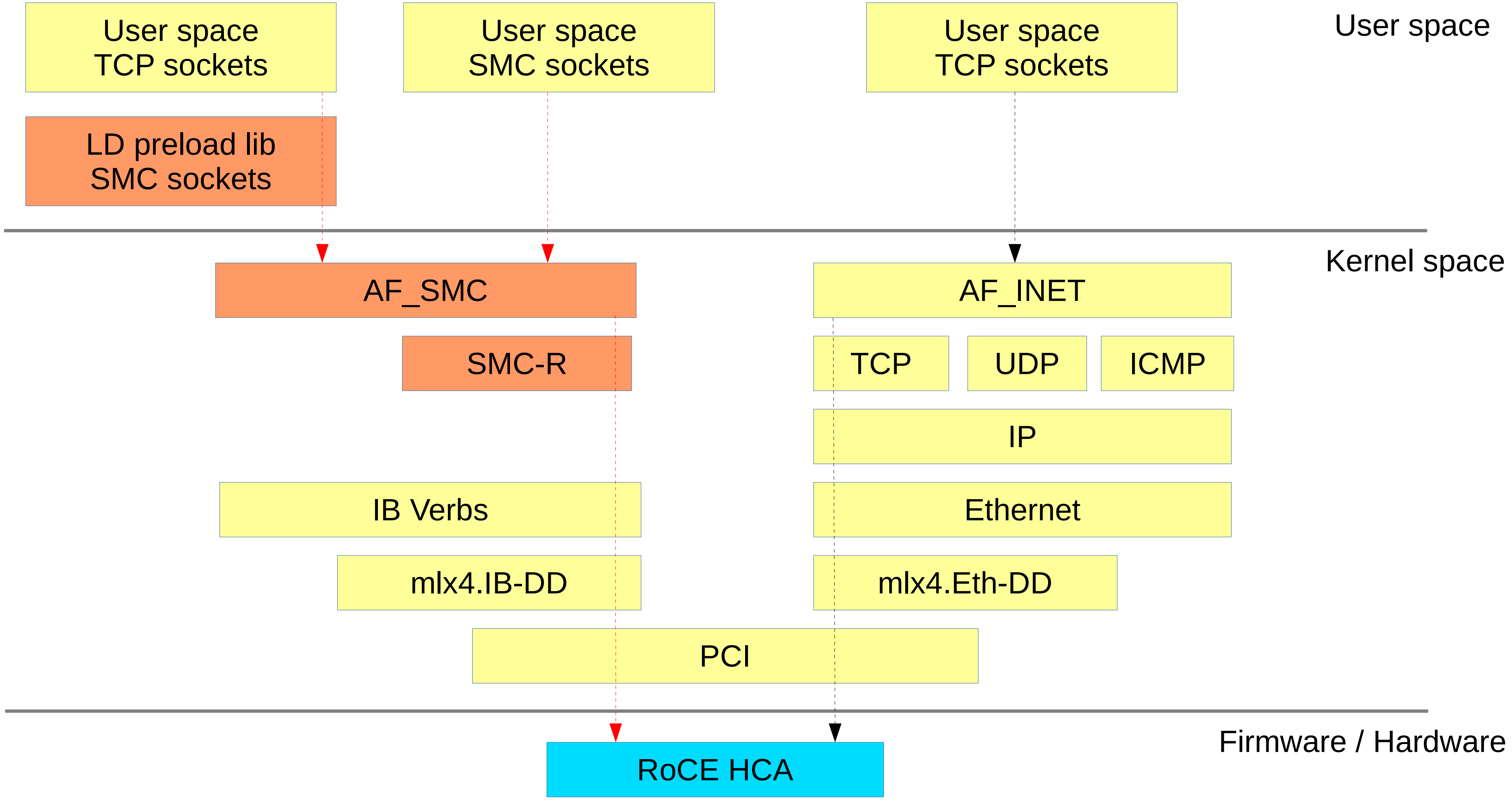
SMC-R concept / overview



RDMA technology provides the capability to allow hosts to logically share memory. The SMC-R protocol defines a means to exploit the shared memory for communications - transparent to the applications!

Linux structure for SMC-R

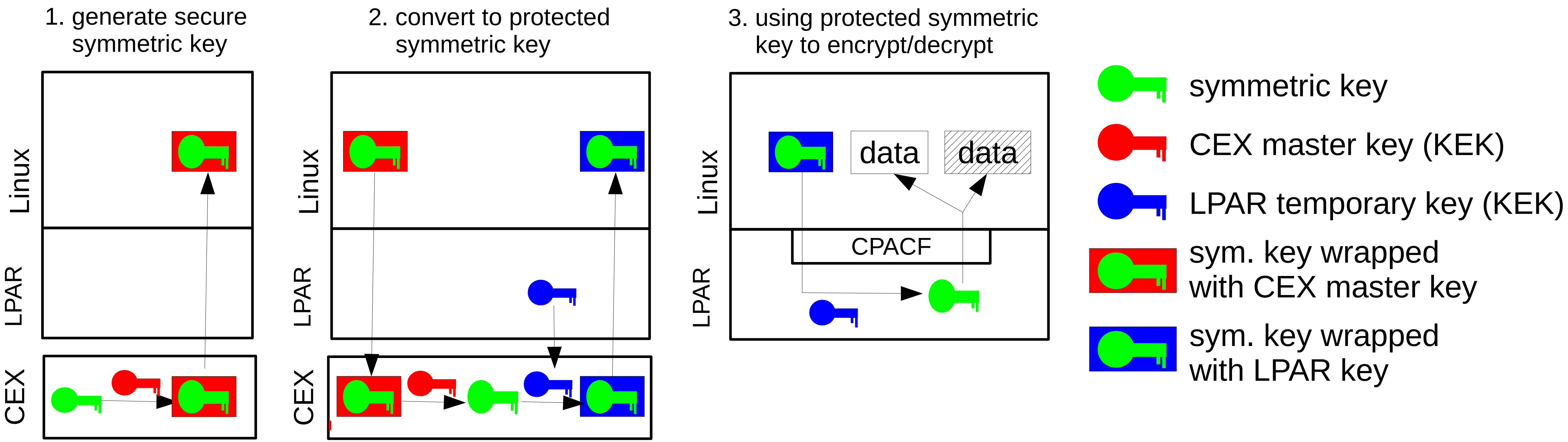
LPAR z/VM



Kernel features – crypto support

LPAR z/VM

- **Protected key encryption for dm-crypt (kernel 4.11)**
  - Consists of the protected key AES module and the secure key API module
  - Allows to encrypt block devices without a clear text key anywhere in memory
  - Userspace tooling for LUKS1 / LUKS2 needs more work, cryptsetup plain works



1. generate secure symmetric key

2. convert to protected symmetric key

3. using protected symmetric key to encrypt/decrypt

symmetric key

CEX master key (KEK)

LPAR temporary key (KEK)

sym. key wrapped with CEX master key

sym. key wrapped with LPAR key

Kernel features: PCI improvements

- **PCI error reporting interface (kernel v4.9)** LPAR z/VM
  - Provide a sysfs interface to allow user space programs to trigger a deconfigure-and-repair action for a specific PCI function
- **PCI I/O TLB flush enhancement (kernel v4.10)** LPAR z/VM
  - Reduce the number of RPCIT instructions in case the hypervisor does not announce that RPCIT can be omitted for invalid -> valid translation-table entry updates

Kernel features - miscellaneous

- **Scatter-gather for AF_IUCV sockets (kernel 4.8)** z/VM
  - Avoid large continuous kernel buffer allocations for AF_IUCV under z/VM
- **Show dynamic and static CPU speed in /proc/cpuinfo (kernel 4.8)** KVM LPAR z/VM
  - Reports the static and dynamic MHz rating of each CPU
- **Add leap seconds to initial system time (kernel 4.8)** KVM LPAR z/VM
  - The current number of leap seconds is a configuration setting of the local machine
  - If the leap seconds have been set correctly they must be subtracted from the TOD clock to determine UTC
- **Performance enhancement for RAID6 gen/xor (kernel 4.9)** KVM LPAR z/VM
  - Speed up the RAID6 syndrome and xor functionsq
- **5 level page tables (kernel 4.11 / kernel 4.13)** KVM LPAR z/VM
  - For x86 machines support for five level of page tables has been introduced with 4.11
  - The z Systems support is planned for kernel version 4.13
    - The user space address limit for z Systems will be 16EB-4KB

Kernel features - miscellaneous

- **IBM z13 specific CPU-MF counter event names (kernel 4.12)** LPAR
  - Add the model specific counter event names of the CPU-Measurement Facility for the IBM z13 machine
  - Allows to use symbolic names instead of the raw event names ‘r[0-9a-z]*’
  - Use ‘lscpumf -C’ for a complete list
- **IBM z13 Multi-Threading CPU-MF counter set (kernel 4.12)** LPAR
  - Add support for the MT-diagnostic counter set introduced with IBM z13
  - Provides access to the counters `MT_DIAG_CYCLES_ONE_THR_ACTIVE` and `MT_DIAG_CYCLES_TWO_THR_ACTIVE`
- **Live patch support (kernel 4.12)** LPAR z/VM KVM
  - Add the architecture backend for z Systems for live patching
  - Provides the basis for the kGraft and kpatch solutions, both allow to update a running kernel with critical patches without a downtime

Linux common code enablements

- **KCOV support (kernel v4.8)** LPAR Z/VM KVM
  - Aka “Kernel coverage information”
  - Exposes kernel code coverage information in a form suitable for coverage-guided fuzzing (randomized testing).
- **UBSAN sanitizer (kernel v4.9)** LPAR Z/VM KVM
  - Aka “Undefined behaviour sanity checker” (e.g. -INT_MIN)
  - Uses compile-time instrumentation to detect undefined behaviours at runtime.
- **CMA support (kernel v4.10)** LPAR Z/VM KVM
  - Aka “Contiguous Memory Allocator”
  - Allows subsystems to allocate big physically-contiguous blocks of memory.

Linux support for IBM z14

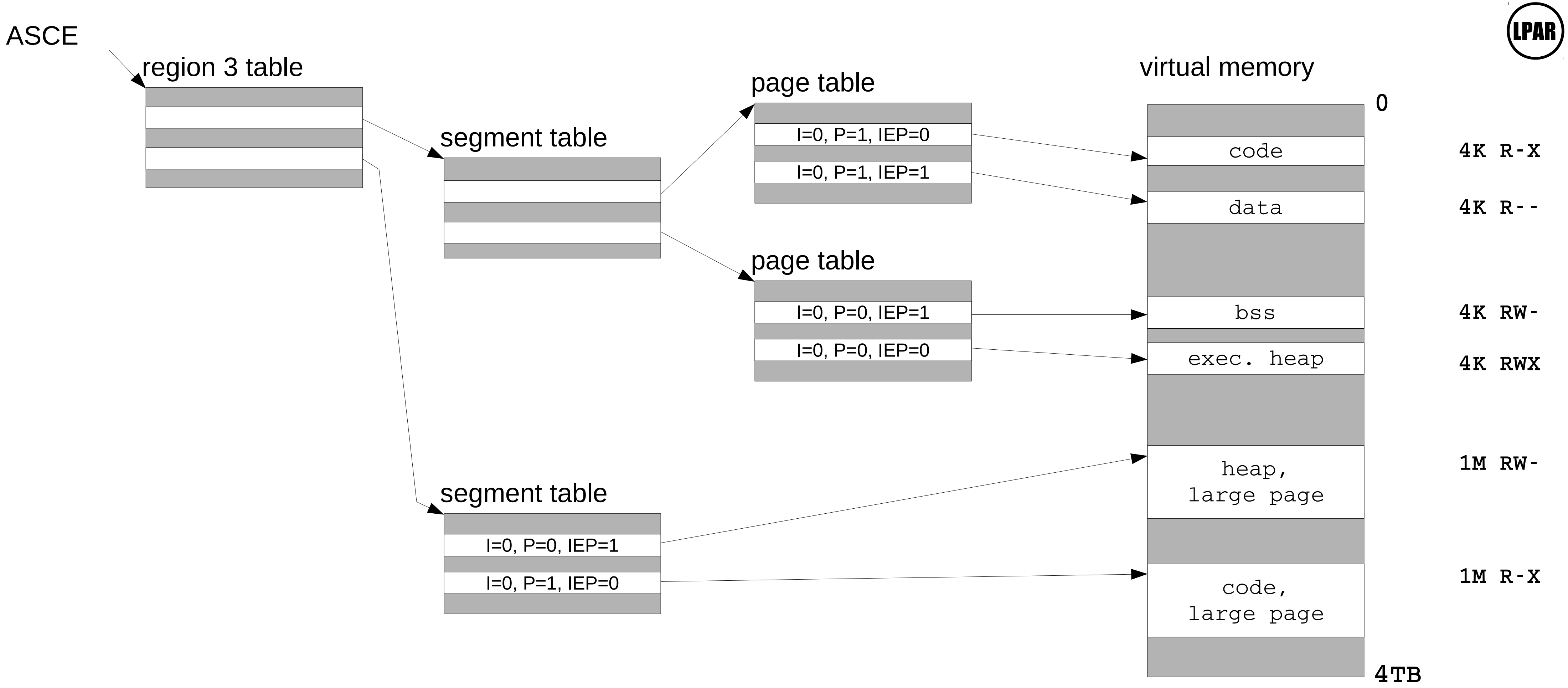
Machine support for IBM z14, partially already upstream with a few more features under development

IBM z14 Support

- **Toleration for Crypto Express 6 cards (kernel 4.10)** LPAR z/VM KVM 7.4
  - Allow to use the new crypto hardware in CEX5 compat mode
- **Report new vector facilities (kernel 4.11)** LPAR z/VM KVM
  - Add two new features flags in /proc/cpuinfo:
    “vxd” for the Vector-Decimal Facility and “vxe” for the Vector-Enhancement Facility 1
  - No additional enablement is required, all vector instructions are enabled by single CR0 bit
- **Instruction execution protection (kernel 4.11)** LPAR z/VM KVM
  - Also know as non-executable mappings or short “noexec”
  - New bits in the segment and page tables to forbid code execution for a 1M segment or a 4K page
  - The PROT_EXEC flag of mmap / mprotect already provides the information which memory regions contains instruction vs. data
  - The presence of the GNU_STACK program header without the execute flag makes all memory mappings with PROT_EXEC==0 to be non-executable

IBM z14 Support: Instruction Execution Protection

LPAR z/VM KVM

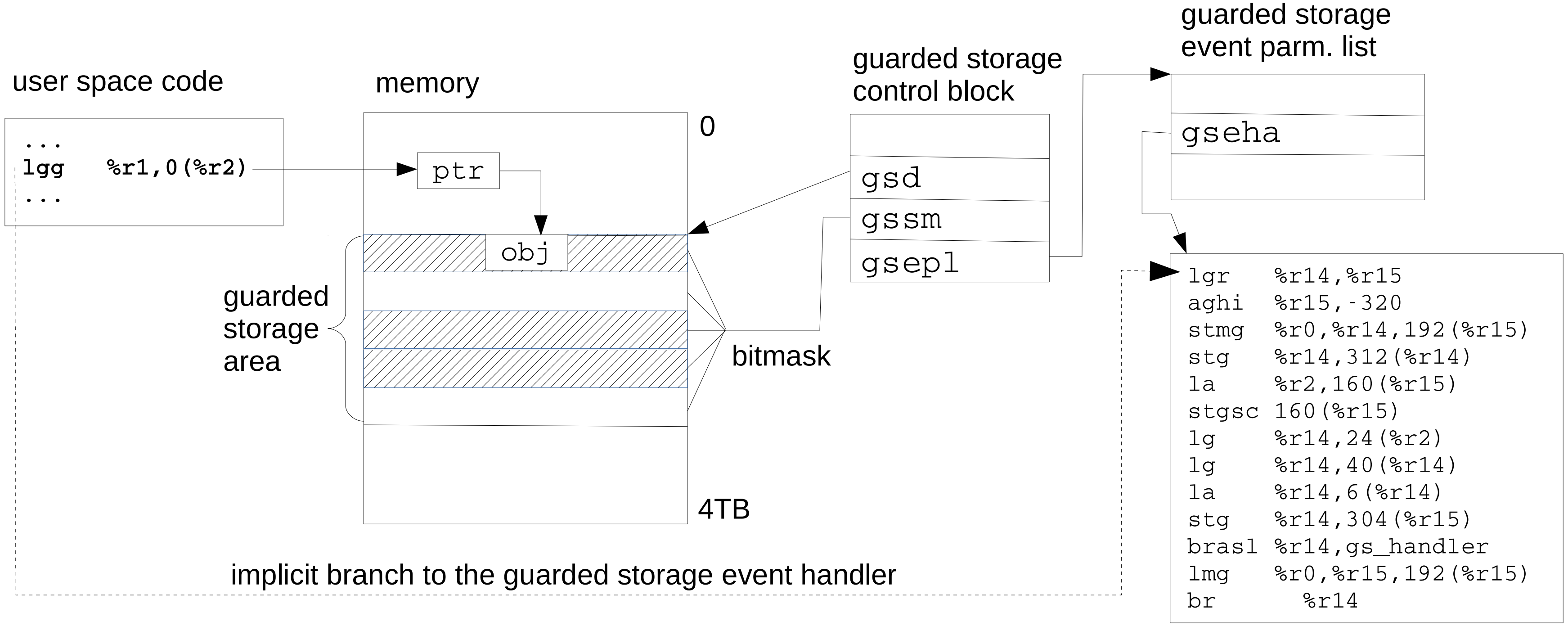


R: PROT_READ, W: PROT_WRITE, X: PROT_EXEC

IBM z14 Support

LPAR z/VM KVM

- **Support for the Guarded Storage Facility (kernel 4.12)**
  - Designed to improve the performance of Java while garbage collection is active
  - Up to 64 regions of memory can be marked as guarded
  - Reading a pointer with the new LGG or LLGFSG instruction will do a range check on the loaded value and automatically invoke a user space handler if one of the guarded regions is affected



user space code
memory
0
ptr
obj
guarded storage area
4TB
bitmask
guarded storage control block
gsd
gssm
gsepl
guarded storage event parm. list
gseha
implicit branch to the guarded storage event handler

```
...
lgg    %r1,0(%r2)
...
```

```
lgr    %r14,%r15
aghi   %r15,-320
stmg   %r0,%r14,192(%r15)
stg    %r14,312(%r14)
la     %r2,160(%r15)
stgsc  160(%r15)
lg     %r14,24(%r2)
lg     %r14,40(%r14)
la     %r14,6(%r14)
stg    %r14,304(%r15)
brasl  %r14,gs_handler
lmg    %r0,%r15,192(%r15)
br       %r14
```

GSE GUIDE SHARE EUROPE Deutsche Region

IBM z14 Support

- **True random number generator (kernel 4.12)** LPAR z/VM KVM
  - The MSA-7 CPACF extension provides a new function for true random numbers
  - Add a hwrng for user space and an arch random function for in-kernel use
- **TOD-Clock Extensions for Multiple Epochs (> kernel 4.13)** LPAR z/VM KVM
  - On September 17, 2042 at 23:53:57.370496 TAI the 64-bit TOD clock will overflow
  - The extended TOD clock format has 8 additional bits, the epoch index
  - Make Linux work with a wrapped 64-bit TOD clock and clock comparators
- **Single-Increment-Assignment Control for memory hotplug (> k. 4.13)** LPAR z/VM KVM
  - Speed up the Attach-Storage-Element SCLP request
  - Improves operation time for some memory hotplug operations

IBM z14 Support

- **Optimized spinlocks with NIAI (> kernel 4.13)** LPAR z/VM KVM
  - Use new sub codes of the NIAI instruction (4, 7 & 8) to reduce cache line traffic
- **TLB flushing improvements (> kernel 4.13)** z/VM KVM
  - Reduce the number of flushed TLB entries for guest-2 TLB flushes (z/VM & KVM)
- **IBM z14 base kernel support (> kernel 4.13)** LPAR z/VM KVM
  - Translate machine type 0x3906 to the ELF platform name “z14”
  - Add the build magic to generate z14-only kernels

s390-tools package – what is it?

- **s390-tools is a package with a set of user space utilities to be used with the Linux on z Systems distributions**
  - It is **the** essential tool chain for Linux on z Systems
  - It contains everything from the boot loader to dump related tools for a system crash analysis
  - Latest version dated 07/28/2017 is 1.39
- **This software package is contained in all major (and IBM supported)** enterprise Linux distributions which support on z Systems
  - RedHat Enterprise Linux version 5, 6, and 7
  - SuSE Linux Enterprise Server version 10, 11, and 12
  - Ubuntu 16.04 Xenial Xerus, 16.10 Yakkety Yak, and 17.04 Zesty Zapus
- **Website:** http://www.ibm.com/developerworks/linux/linux390/s390-tools.html
- **Feedback: linux390@de.ibm.com**

s390-tools package – the content

zipl
Boot

chccwdev
chchp
chcpumf
chiucvallow
chreipl
chshut
chzcrypt
chzdev
cio_ignore
~~chmem~~
Change

zkey
Crypto

dasdfmt
dasdinfo
dasdstat
dasdview
fdasd
tunedasd
DASD

lscss
lschp
lscpumf
lsctrdef
lsdasd
lshmc
lshvciucv
lsiucvallow
lsluns
lsqeth
lsreipl
lsscm
lsshut
lstape
lszcrypt
lszdev
lszfcp
~~lsmem~~
Display

hmcdrvfs
zdsfs
Filesystem

cpacfstats
cpacfstatsd
mon_fsstatd
mon_procd
ziomon
hyptop
Monitor

ip_watcher
osasnmpd
qetharp
qethconf
qethqoat
start_hsnc
xcec-bridge
znetconf
Network

tape390_display
tape390_crypt
Tape

vmconvert
vmcp
vmur
cms-fuse
z/VM

cpuplugd
iucvconn
iucvtty
ts-shell
ttyrun
Misc

Dump
&
Debug
dbginfo
dumpconf
dump2tar
zfcpdump
zfcpdbf
zgetdump
scsi_logging_level

s390-tools package – lszdev / chzdev

- Example output for lszdev

```
# lszdev
TYPE         ID                          ON   PERS  NAMES
dasd-eckd    0.0.0190                    no   no
dasd-eckd    0.0.0191                    no   no
dasd-eckd    0.0.019d                    no   no
dasd-eckd    0.0.019e                    no   no
dasd-eckd    0.0.0592                    no   no
dasd-eckd    0.0.6527                    yes  no    dasda
dasd-eckd    0.0.6528                    yes  no    dasdb
qeth         0.0.f500:0.0.f501:0.0.f502  no   no
qeth         0.0.f5f0:0.0.f5f1:0.0.f5f2  yes  no    eth0
generic-ccw  0.0.0009                    yes  no
generic-ccw  0.0.000c                    no   no
generic-ccw  0.0.000d                    no   no
generic-ccw  0.0.000e                    no   no
generic-ccw  0.0.0600                    no   no
```

util-linux – the content

chcpu
chmem
lscpu
lsmem
wdctl
zramctl

CPU & memory

agetty
cal
ctrlaltdel
dmesg
getopt
ldattach
logger
namei
script
scriptreplay
setterm
tailf

Shell & Terminal

chrt
ionice
ipcmk
ipcrm
ipcs
kill
linux32
linux64
lsipc
lsns
nsenter
prlimit
renice
runuser
s390
s390x
setarch
setpriv
setsid
taskset
uname26
unshare

Process

col
colcrt
colrm
column
hexdump
look
more
rev
ul

Input & output

last
lastb
login
lslogins
mcookie
mesg
nologin
su
sulogin
utmpdump
wall
write

User & Login

fallocate
findfs
findmnt
flock
fsck
fsfreeze
fstrim
isosize
losetup
lslocks
mkfs
mount
mountpoint
pivot_root
rename
switch_root
umount
whereis
wipefs

File-system

adpart
blkdiscard
blkid
blockdev
cfdisk
delpart
eject
fdisk
lsblk
mkswap
partx
raw
rawdevices
resizepart
sfdisk
swaplabel
swapoff
swapon

Disk & media

IBM Knowledge Center

- **The central location for finding and organizing information about IBM products**
- **How to get there:**
  - Search for “IBM Knowledge Center” or go directly to
  - https://www.ibm.com/support/knowledgecenter/
- **How to get to Linux on z Systems stuff:**
  - Search for “Linux z” within IBM Knowledge Center or go directly to
  - https://www.ibm.com/support/knowledgecenter/linuxonibm/liaaf/lnz_r_main.html
- **Highlights**:
  - Mobile enabled
  - Not only pdf, but also full text view and search
  - Classified by topics
  - Direct links to related information like Redbooks, Whitepapers,…

Linux on IBM z System reference

Linux on z Systems (official):
http://www-03.ibm.com/systems/z/os/linux/index.html
Linux on z Systems (technical):
http://www.ibm.com/developerworks/linux/linux390/index.html
z/VM:
http://www-03.ibm.com/systems/z/solutions/virtualization/zvm
IBM Wave for z/VM:
http://www-03.ibm.com/systems/z/solutions/virtualization/wave
IBM Blockchain:
https://www.ibm.com/blockchain/index.html
IBM DB2 for Linux:
https://www.ibm.com/analytics/us/en/technology/db2/db2-linux-unix-windows.html
IBM Spectrum Scale:
http://www-03.ibm.com/systems/storage/spectrum/scale/index.html
Hardware cryptographic support for IBM Z and IBM LinuxONE with Ubuntu Server:
https://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102721
Hardware cryptographic support of IBM z Systems for OpenSSH in RHEL 7.2 and SLES 12 SP1:
https://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102653

Questions?