**Symmetric keys are used with symmetric algorithms (Data Encryption Standard or DES); Triple DES or TDES and Advanced Encryption Standard (AES) and both parties (the encryptor and decrypter) must have a copy of the key, which must be a secret between the two.**

* Anyone, and everyone, who has a copy of the symmetric key can decrypt the data encrypted with that key.

**Asymmetric algorithms or Public Key Architecture (PKA) use a key pair to protect data.**

* Don’t confuse a clear key’s secure key with public keys used by asymmetric algorithms.
  - With PKA, two different, but mathematically related keys are used.
  - The public key is made available publicly and can be used by anyone who wishes to send data securely to the owner.
  - The data that has been encrypted using a public key can only be decrypted using the corresponding private key.
  - Anyone who has a copy of the public key can encrypt their own data to send, but they cannot use that public key to decrypt data encrypted by the owner of the private key.
  - Since the private key must be used to decrypt the data encrypted by the public key, that private key should be well protected and only available to the owner of the public key.

# Cheatsheet #60 zTidBits

## System z Cryptographic Hardware

- System z cryptographic hardware supports four cryptographic capabilities:
  1. Data confidentiality (encrypting/decrypting data using symmetric and/or asymmetric algorithms)
  2. Message integrity (message authentication, modification detection, nonrepudiation)
  3. Financial functions (using symmetric algorithms to protect PINs associated with credit cards or financial transactions)
  4. Key management (security and integrity of keys)

### Clear Key vs Secure Key vs Protected Key

<table>
<thead>
<tr>
<th><strong>Clear Key</strong></th>
<th><strong>Secure Key</strong></th>
<th><strong>Protected Key</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Crypto Hardware can use either clear key, secure key or protected key and no matter which you choose, there is no difference between the cryptographic key or the key name.</td>
<td>IBM Crypto Hardware can use either clear key, secure key or protected key and no matter which you choose, there is no difference between the cryptographic key or the key name.</td>
<td>IBM Crypto Hardware can use either clear key, secure key or protected key and no matter which you choose, there is no difference between the cryptographic key or the key name.</td>
</tr>
</tbody>
</table>

### NOTE

- The difference is the protection provided for the key value that protects the data.
  - The secure hardware includes tamper detecting technology to protect against attacks involving probe penetration, power supply, and temperature rise. This is done with the aid of a temperature sensor, and using that information, the device is designed to protect the data.
  - If a tamper is detected the circuitry will “zeroize” the card wiping out the key so they cannot be compromised.
  - When a clear key is created, that key is encrypted under a master key, and the underlying key value is never explicitly exposed in the clear, outside of the secure hardware.
  - When a clear key needs to leave the secure hardware (for example to be stored in a dataset like the CKDS) the encrypted version of the key is stored.

### NOTE

- Transport purposes, a key can be encrypted under a key-encrypting key instead of the master key, but no in this instance will a secure key exist in the clear outside of the secure hardware.
- Clear keys may exist on the workstation process, or it may exist in an address space during key entry or in use by an application. When the network, 2009 LEC, the z2 now supports protected keys that does not rely on the transport resistant secure hardware, but the protected key is encrypted under a wrapping key until it is used within the CPACF which provides better performance than the secure hardware.
- LPAR
  - A protected key is an operational key that is encrypted under a wrapping key associated with the LPAR.
  - When the protected key is brought into the CPACF it is unwrapped and the clear value is used to perform the crypto operation.

### NOTE

- Protected keys can be DES (Data Encryption Standard), TDES (Triple DES), or AES (Advanced Encryption Standard).
- A secure key never exists in the clear outside the tamper resistant boundary of the secure hardware.
- A protected key never exists in the clear outside the tamper resistant hardware.
- A secure key is encrypted under a master key when it leaves the tamper resistant hardware.
- A clear key is encrypted under the key-encrypting key that is written each time the LPAR is activated or reset through the HMC or Support Element (SE).

### NOTE

- The key wrapping is stored in the Hardware Security Storage Area (HSSA) – see #32 zTidBits (Free Storage – more WAS).

### NOTE

- Some additional costs associated with the CPACF wrapping is not applicable to an operating system or application service, but it does not protect the tamper resistant hardware technology.

### NOTE

- There are two variations of the wrapping key: one for DES/TDES keys and one for AES keys.

### NOTE

- There are additional costs associated with secure keys and additional CPU costs, but customers must make a business decision about whether their security requirements warrant the cost of secure key support.

### Clear Keys

- Clear keys provide the best performance of the three types because their operations are done on the general purpose processor. Security is achieved by safely storing the key on the PCIe card of the Self-Timed Interface, so effectively you’re executing an I/O operation to get the data and keys out to the card.

### Secure Keys

- Secure keys are stored in the secure hardware module. While the crypto work is offloaded to the card, there is still some CPU costs in getting the work formatted for and routed to the card and then in receiving the results back from the card and passing those results back to the caller.

### Protected Keys

- Protected keys fall in between clear keys and secure keys in terms of performance.
  - Performance is closer to that of clear keys, although they do have some additional overhead.

## Cheatsheet #60 zTidBits

### Export Restrictions

The U.S. Government considers encryption technology to be a munitions, and accordingly requires strict controls of the ability to export the technology. The sufficiency includes Cryptographic technology within the machine, IBM controls access to the cryptographic hardware via microcode, and the U.S. Government limits where that microcode can be exported. All of the installed crypto z2 crypto device to the appropriate microcode to be installed and operational before the export restrictions can be used. On the z890/290, z9 and z10 machines, this microcode is ordered as no-charge Feature Code $3883. On the z900 machines, the microcode will be shipped from the machine. IBM software that implements encryption will also check for the presence of this feature code before performing encryption in software.

### Cryptographic Software and Integrated Cryptographic Service Facility, ISCF, is the software that provides the interface to the hardware. As new functions are implemented in the hardware, new versions of ICSC will be available to invoke those functions. ICSC is available as a component of zOS, however the current versions are available via web download.

### Crypto APIs

- There are a few crypto APIs that are made available directly to an application, but most of the crypto hardware can only be accessed by using the cryptographic Application Programming Interfaces (APIs).

### The ICSF Application Programmer’s Guide provides a table for each API that describes the hardware required to support the API located on RESOURCSE LINK.

### ICSF supports APIs for applications programs written in a number of high-level languages as well as assembler.

### The high-level language versions are: C, COBOL, FORTRAN, PL/I.

### How the approach is coded for a machine that uses the symmetric hardware is used.

### The application controls which APIs or cryptographic instructions are invoked, and what parameters are passed via the API.

### Specific parameters may be encrypted on a particular cryptographic hardware device, but not on another and therefore, the parameters can impact how the work is routed.

### Some applications may perform the cryptographic functions in their own address spaces, but others may have the software or ICSF.

### NOTE

- Invoking the API can be expensive, some IBM software will only invoke ICSF and not the hardware if the amount of data to be encrypted is large enough to overcome the overhead of passing the data to ICSF.

### Several reasons to move crypto function off the system board and into the I/O cage

1. **Availability:** Adding crypto functionality to a processor requires an output of the entire machine PINs, and no data is free from attack until security is introduced into the system at the I/O cage.
2. **Scalability:** depending on the platform and other cards installed in the I/O cage, up to eight cryptographic features can be installed in a processor. As workload increases, additional features can be ordered on and installed, without an outage to the LPAR.
3. **Cutting the crypto functions off the processor (the I/O cage and physical attacks) to the PCI cards allowed IBM to better manage engineering cost of this additional protection.**
4. **The z890 Dx3 (10/09) included support for a new PCI card, the Crypto Express3 and support in the CPACF for the new protected key function key encryption supporting applications as well as making it easier to manage the crypto devices.
5. **The z900 three crypto hardware cards are all z10:** The OP Assist for Crypto Configuration (CPACF) is integrated into the PU in the host, and the Crypto Express4 and Crypto Express3 are PCI devices installed in the I/O cage.

### NOTE

- On the z10 (and the z9), a crypto engine can be dynamically changed (from accelerator to coprocessor or coprocessor to accelerator) via the Hardware Management Console (HMC) without taking an outage or ICSF. The operating system or the LPAR. This means that as workload changes you can reconfigure the crypto devices to take advantage of the performance and throughput characteristics of each without incurring an outage.