z/TPF Communications and Security Enhancements

2023 TPF Users Group Conference
April 24-26, Dallas, TX
Communications Subcommittee

—

Jamie Farmer

IBM

Agenda

- z/TPF TCP/IP network override support
- Elliptic curve cryptography (ECC) support for secure network connections
- z/TPF cryptographic inventory
- z/TPF TCP/IP performance enhancements
- Support for latest cryptographic hardware on IBM z16

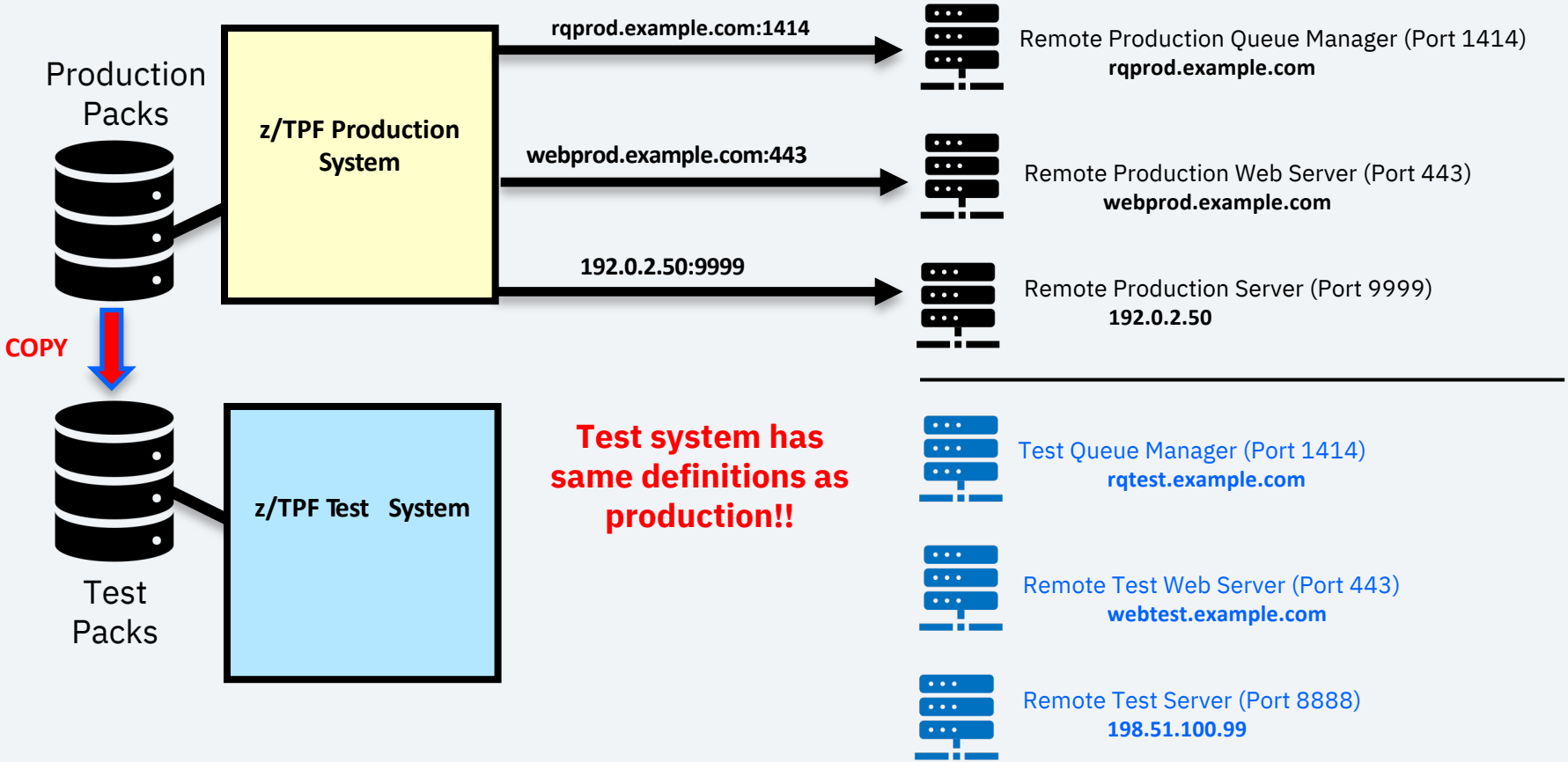
Agenda

- **z/TPF TCP/IP network override support**
- Elliptic curve cryptography (ECC) support for secure network connections
- z/TPF cryptographic inventory
- z/TPF TCP/IP performance enhancements
- Support for latest cryptographic hardware on IBM z16

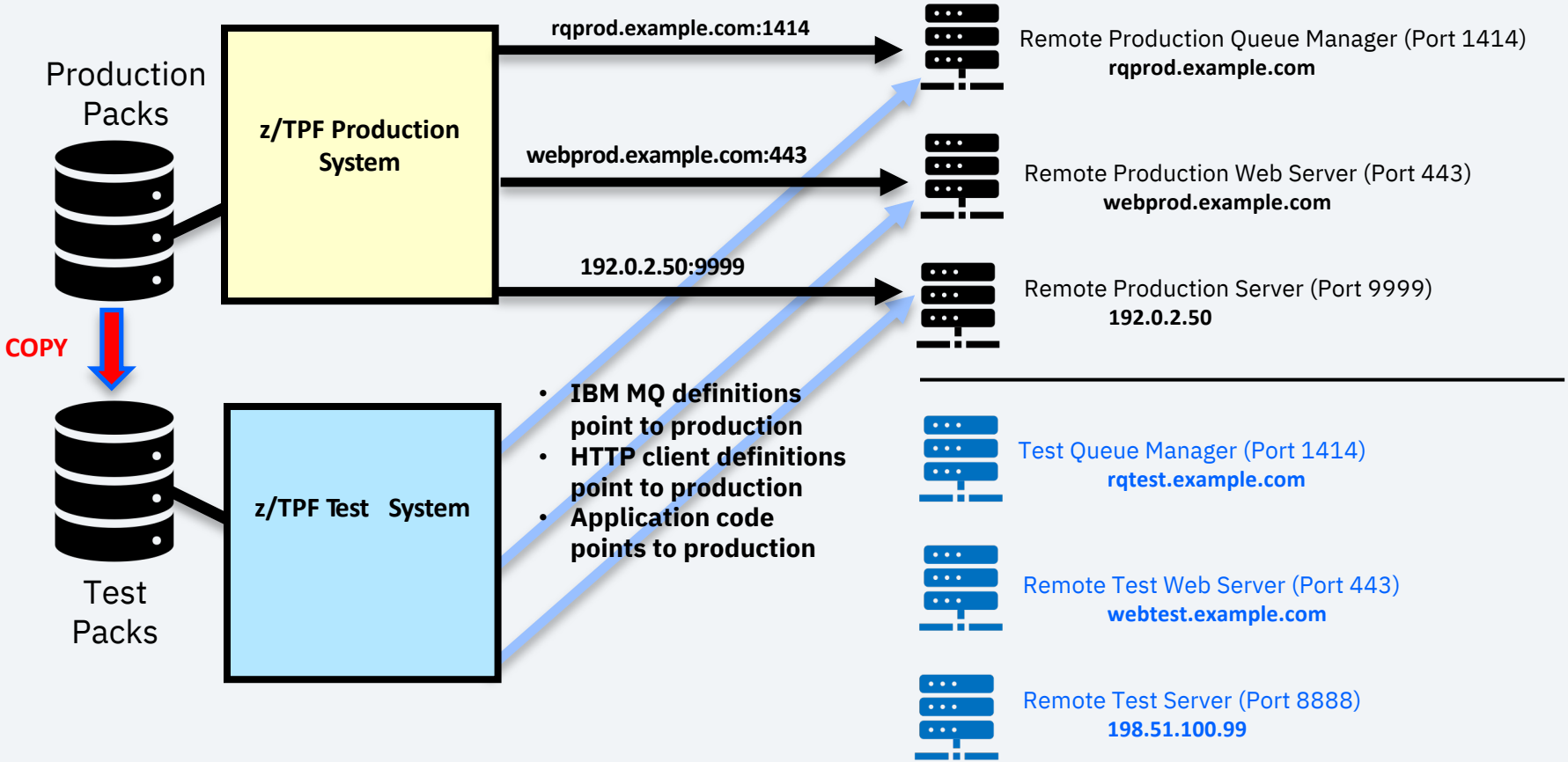
TCP/IP Network Override -- Problem Statement

- Many of you make a copy of a production system as a base for test systems.
  - A physical copy of the entire system is made
    - Customer databases, saved configuration information, etc.
- The copied test system is then modified to…
    - Scrub sensitive data from customer databases
    - Redefine IP connectivity for the system
    - Update configuration to prevent access to remote production servers.
      - For example, delete IBM MQ channels and redefine them for test.

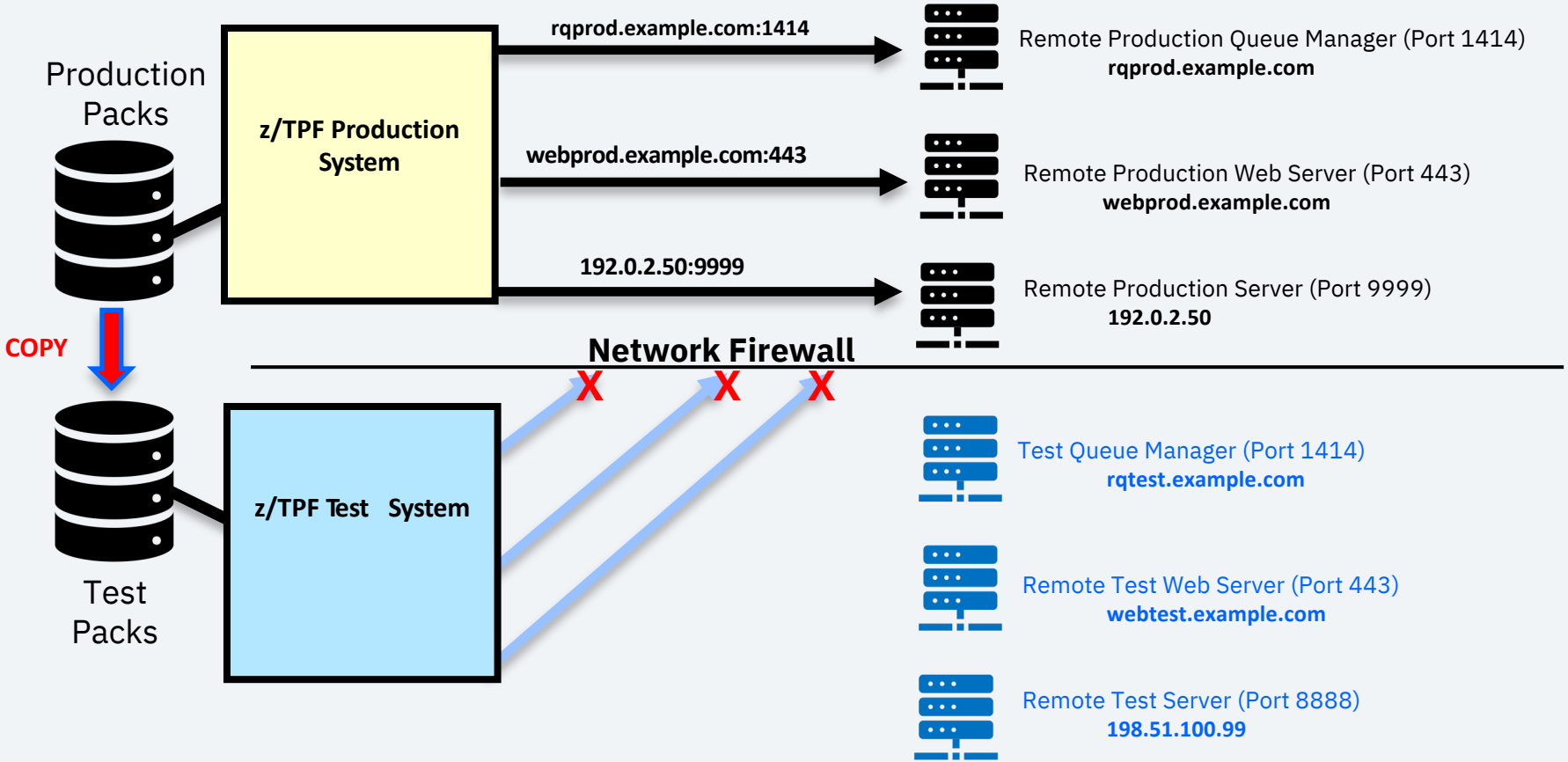
Test Network Override -- Problem Statement Example



Test Network Override -- Problem Statement Example



Test Network Override -- Problem Statement Example



As-Is: Copying a Production System to Test

- When deploying a test system that was copied from production, definitions must be updated to connect to test servers. For example,
  - IBM MQ definitions
  - High-speed connection configuration files
  - HTTP client configuration files
  - REST consumer OpenAPI documents
  - FTP client profiles
  - Java property files with remote hostnames or IP addresses
  - Application code connecting directly to production servers

- Updating all applications and middleware that access remote production systems is time consuming and error prone!

To Be: Introducing TCP/IP Network Override

- With TCP/IP network override support, you to remap remote connectivity and make it easier to redirect outbound z/TPF traffic to test servers.

- Use a command to override the following remote definitions:
  - Production IP Address -> Test IP Address
  - Production IP Address and Port -> Test IP Address and Port
  - Production Hostname -> Test Hostname
  - Production Hostname -> Test IP Address

TCP connect API

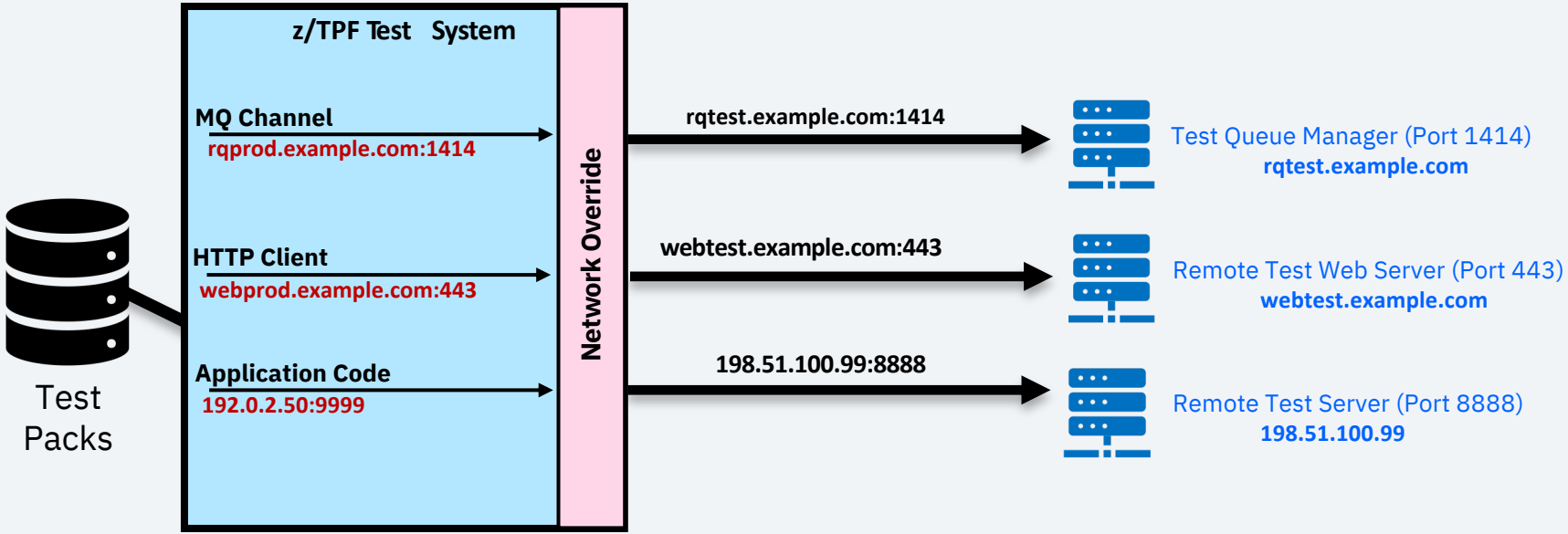
gethostbyname API

- For example:

ZTTCP OVERRIDE HOST- rqprod.example.com NEWHOST- rqtest.example.com
ZTTCP OVERRIDE HOST- webprod.example.com NEWHOST- webtest.example.com
ZTTCP OVERRIDE IP- 192.0.2.50 PORT-9999 NEWIP-198.51.100.99 NEWPORT-8888

- The defined overrides take effect immediately and persist across an IPL.

To Be: TCP/IP Network Override

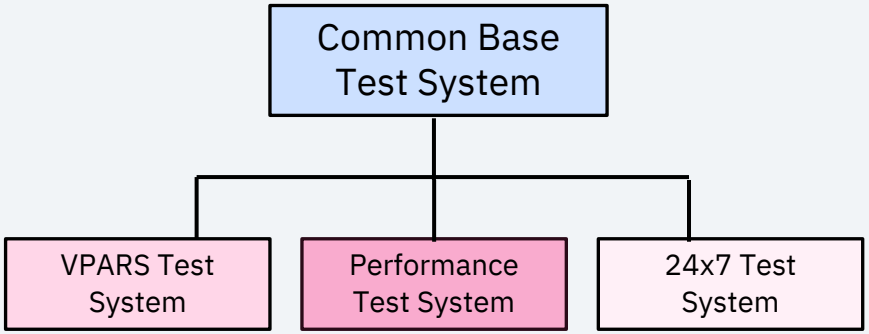


To Be: Displaying and Validating the Override Table Usage

```
User:  ZTTCP OVERRIDE DISPLAY ALL

TTCP0079I 08.32.34 DISPLAY NETWORK OVERRIDE ENTRIES

DISPLAY HOSTNAME OVERRIDE ENTRIES
  HOST-rqprod.example.com
       NEWHOST-rqtest.example.com
       USED-7
  HOST-webprod.example.com
       NEWHOST-webtest.example.com
       USED-19

DISPLAY IP ADDRESS ENTRIES
  IP-192.0.2.50 PORT-9999 NEWIP-198.51.100.99 PORT-8888
     USED-43

END OF DISPLAY
```

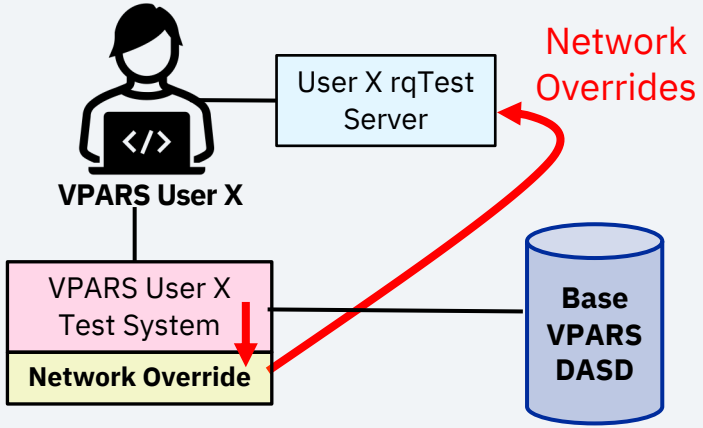
Other Test Usage

Deployment of Test Systems



Each can be deployed with varying z/TPF overrides

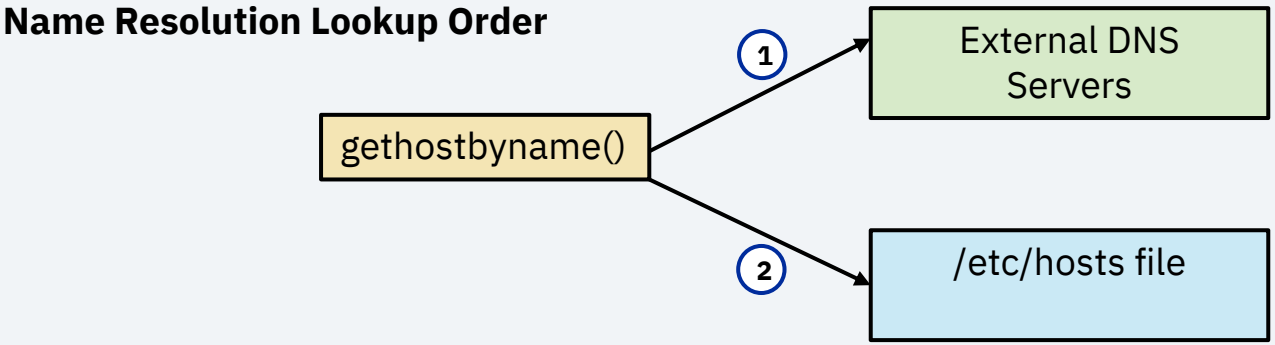
Individual Test Users



In the z/TPF lab we have started using network overrides on our test systems.

As-Is: Overriding DNS Hostnames in Production

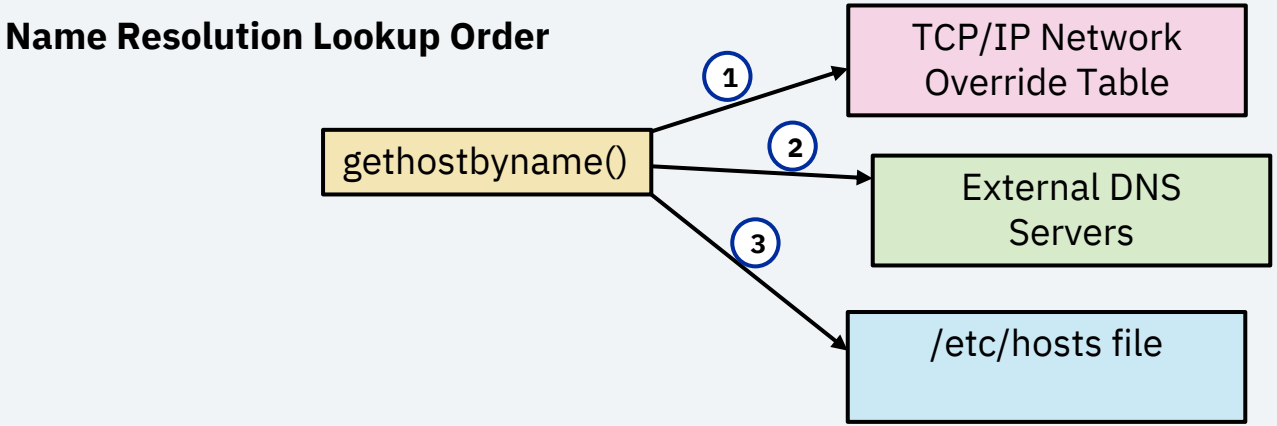
- In some cases, users do not want to use external DNS to resolve a hostname on production.



- Updating /etc/hosts doesn't bypass external DNS lookups but acts as a secondary lookup.

To Be: Overriding DNS Hostnames in Production

- The TCP/IP network override is highly efficient and can be used on production systems.



- TCP/IP network override support provides true overrides for external DNS servers!

Value Statement

Having a system-wide TCP/IP network override reduces the time it takes to deploy z/TPF test systems, improves the testing capabilities of TPF, and allows users to override their external DNS.

Delivered with PJ46729 (Aug 2022)

Agenda

- z/TPF TCP/IP network override support
- **Elliptic curve cryptography (ECC) support for secure network connections**
- z/TPF cryptographic inventory
- z/TPF TCP/IP performance enhancements
- Support for latest cryptographic hardware on IBM z16

Elliptic Curve Cryptography -- Problem Statement

- The industry is moving away from using RSA for exchanging the secret symmetric key of a TLS session to using an ephemeral public-private key pair.
  - Ephemeral means the public-private key pair used to exchange the secret symmetric key of an OpenSSL session is uniquely generated for each session
  - Ephemeral public-private keys provide ‘Perfect Forward Secrecy”
    - Limits the exposure if a private key is somehow compromised

- Added ephemeral Diffie-Hellman (DHE_*) ciphers to z/TPF OpenSSL in Dec 2020 (APAR PJ46292)
    - Operations are performed only in software – expensive!

Elliptic Curve Cryptography -- Hardware Acceleration

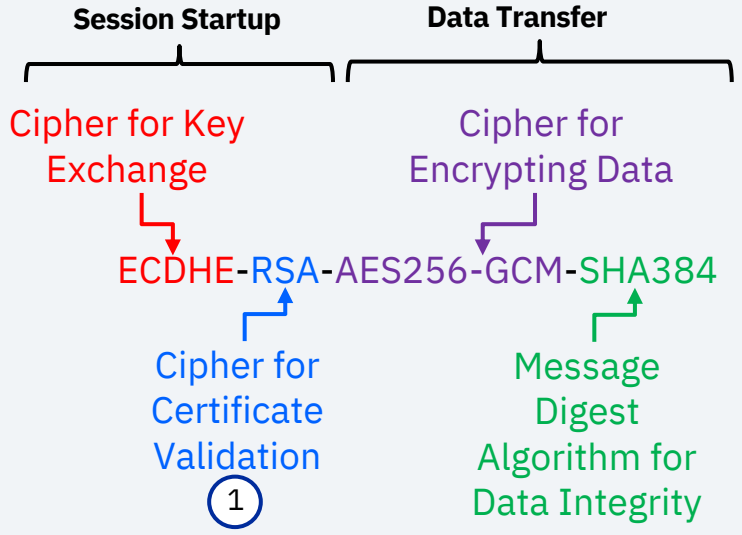
- The IBM z15 processor provides hardware acceleration to perform scalar multiply operations
  - Scalar multiply is the most expensive computational component of ECC when starting TLS sessions. Is used when …
    - Creating the ephemeral ECC public-private key pair
    - Deriving the secret symmetric key to use for the TLS session

- The hardware acceleration is performed in the CPACF hardware
  - Same on-chip coprocessor used to perform AES, SHA, etc.

```
CPAC0012I 11.27.36 CPACF QUERY DISPLAY
   SHA-1:         ENABLED
   DES/TDES:      ENABLED
   AES-128:       ENABLED
   SHA-256:       ENABLED
   AES-256:       ENABLED
   SHA-512:       ENABLED
   DRNG:          ENABLED
   TRNG:          ENABLED
   AES-128-GCM:   ENABLED
   AES-256-GCM:   ENABLED
   SHA-384:       ENABLED
   ECC:           ENABLED
```

To-Be: Elliptic Curve Cryptography on z/TPF

- The following OpenSSL TLS ciphers have been added for z/TPF:
  - ECDHE-RSA-AES128-SHA256
  - ECDHE-RSA-AES256-SHA384
  - ECDHE-RSA-AES128-GCM-SHA256
  - ECDHE-RSA-AES256-GCM-SHA384
- Use ephemeral ECC for key exchange, but still use RSA for certificate authentication and validation.
- Operations in ECC will be hardware accelerated when on an IBM z15 or higher
  - IBM z14 and below, ECC operations will be done in software making it more expensive
- ECDHE ciphers are one of the few supported in TLS 1.3 for key exchange.



1 **Performed in CryptoExpress Card, everything else in CPACF**

Crypto Express cards are still required for certificate validation using RSA

To-Be: Elliptic Curves on z/TPF

- When establishing a TLS session using ECDHE_*, the elliptic curve to use is also negotiated along with the cipher.
- The following curves are supported on z/TPF with hardware acceleration
  - X25519
  - P256
  - X448
  - P384
  - P521
  - NIST Curves
  - Montgomery Curves
- The OpenSSL default curve list, in preference order, is X25519:P256:X448:P521:P384
  - You can modify the curve list.
    - For example, your business states that only NIST curves should be used.

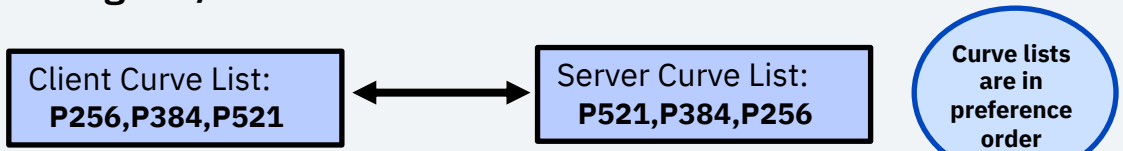
To-Be: Changing the z/TPF Default Curve List

- Use new options on the ZSSLD command to modify the z/TPF default curve list
  - Takes effect for ALL new sessions for ALL applications on ALL processors of the loosely coupled complex.
    - Information saved in a Format-2 Global record

- For example:

```
User: ZSSLD DEFCURVE DEFINE-P-256:P-521:P-384
System: SSLD0012I 12.33.51 THE P-256:P-521:P-384 USER-DEFINED ECC CURVE LIST IS SUCCESSFULLY DEFINED.
```

Note: For z/TPF servers, use the SSLSERVP option in ZNKEY to honor the server's priority over the client connecting to z/TPF.



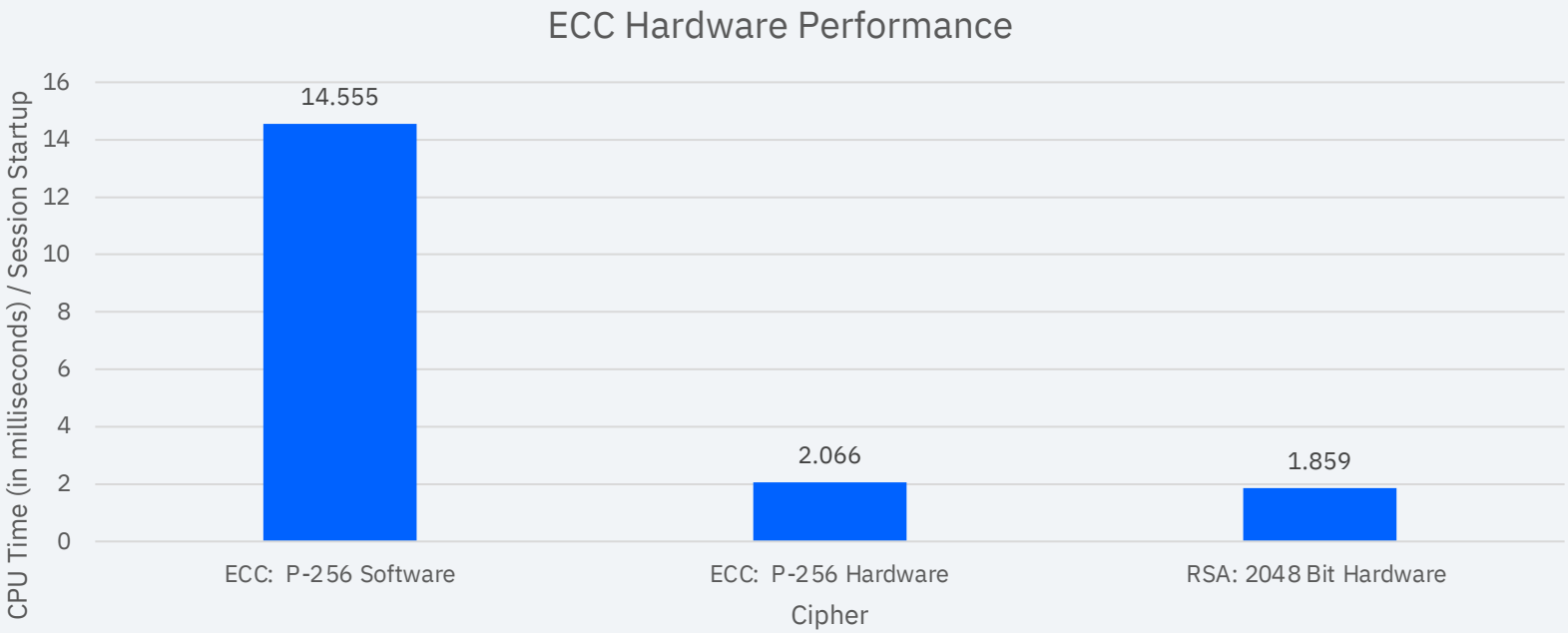
SSLSERVP=YES: P521 would be selected
SSLSERVP=NO : P256 would be selected

To-Be: Network Compliance Updates for ECC Curves

- New fields in the ZDCOM network compliance display identify which curves are allowed and used for each port on the system.

```
DCOM0017I 16.26.14 NETWORK COMPLIANCE SERVER DISPLAY FOR PORT 7510
 PORT-7510       MODEL-SERVER    TLS-Y
 PROTO-TCP       NAME-SSL7510
 TLS INFORMATION:
  TLS VERSIONS USED   : TLS 1.2
  TLS VERSIONS ALLOWED: TLS 1.2 _
  TLS CIPHERS USED    : ECDHE-RSA-AES128-GCM-SHA256
  TLS CIPHERS ALLOWED : ECDHE-RSA-AES128-GCM-SHA256
  TLS ECC CURVES USED   : P256
  TLS ECC CURVES ALLOWED: P256, P384, P521
  PRIVATE KEY KEYSTORE NAME   : TPF2048
  SERVER PRIVATE KEY SIZE     : 2048
 TPF CERTIFICATE:
  PUBLIC KEY LENGTH  : 2048
  SIGNATURE ALGORITHM: sha256WithRSAEncryption
  SUBJECT INFO       : /C=US/ST=New York/L=pok/O=ibm/OU=TPF/CN=tpf.example.com/
                       emailAddress=myemail@example.com
  ISSUE DATE         : Jun 24 14:17:09 2021 LST
  EXPIRATION DATE    : Nov 08 14:17:09 2048 LST
 CLIENT AUTHENTICATION: NO
END OF DISPLAY
```

To-Be: Elliptic Curve Performance



- **Performance is comparable to 2048 RSA key exchange in hardware and the P-256 ECC operation provides stronger security (P256 is equivalent to 3072 bit RSA).**
- **Very expensive in software!**

Value Statement

With ephemeral elliptic curve ciphers, the network security of the z/TPF system is improved and satisfies the requirement of perfect forward secrecy.

Delivered with PJ46719 (Jun 2022)

Agenda

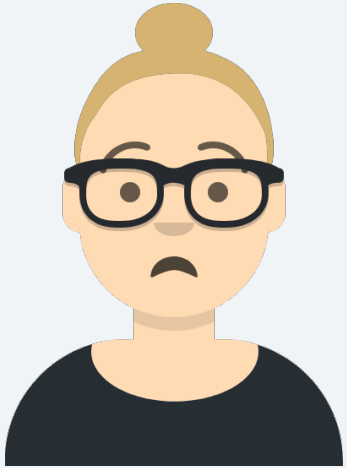
- z/TPF TCP/IP network override support
- Elliptic curve cryptography (ECC) support for secure network connections
- **z/TPF cryptographic inventory**
- z/TPF TCP/IP performance enhancements
- Support for latest cryptographic hardware on IBM z16

Secure Key Inventory -- Problem Statement

- Obtaining z/TPF compliance information for audit purposes is time consuming.
- In 2021, the z/TPF lab delivered network compliance tooling to identify compliance information for data in flight (across the network)
- Identifying the usage of secure keys to protect data still requires significant time and effort.
  - For example, you need to manually identify usage of secure keys on z/TPF

As-Is: Identifying At Rest Encryption on z/TPF

- Christine, a Security Administrator for the z/TPF system, is asked to identify all the data at rest that is encrypted on z/TPF and which databases are not.
  - Not only is she required to prove it is encrypted, but she needs to also identify how it's being encrypted – which cipher algorithms?
- Christine knows this will be time consuming as the use of secure keys on z/TPF is spread across multiple application programs including usage in z/TPFDF.

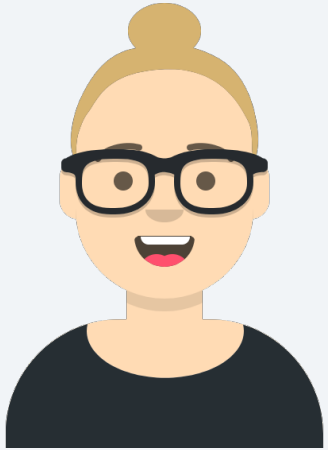


Christine
Security
Administrator

To-Be: Identifying z/TPFDF Database Encryption

- With a simple command, Christine can easily identify which z/TPFDF databases are encrypted or more importantly not encrypted.
  - New command also provides filter options including the ability to write information to a file on the file system.

```
ZUDFM ENCRYPT USERSUM
CSMP0097I 07.46.15 CPU-B SS-BSS  SSU-HPN  IS-01
UDFM0628I 07.46.15 ENCRYPTION SUMMARY INFORMATION FOR USER-DEFINED z/TPFDF
FILES
FILE ID  DBNAME    ENCRYPTION  KEY NAME  CIPHER          DIGEST
-------  --------  ----------  --------  --------------  ------
B071     SCHED     N           NONE      NONE            NONE
B211     CREDIT1   Y           DFCREDKY  AES256CBC       SHA256
B212     CREDIT2   Y           DFCREDKY  AES256CBC       SHA256
B221     FLIGHT1   Y           DFFLTKEY  AES128CBC       NONE
B222     FLIGHT2   Y           DFFLTKEY  AES128CBC       NONE
END OF DISPLAY
```



Christine
Security
Administrator

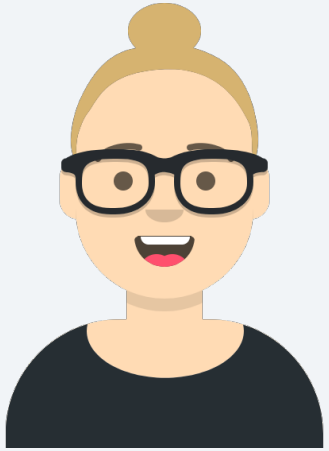
The auditor was concerned that the SCHED database is not secure, until Christine explained that this information is publicly available and does not require data to be encrypted.

Note: This display filters out IBM-Shipped z/TPFDF files

To-Be: Identifying Usage of Secure Keys on z/TPF

- With a simple command, Christine can also identify all the application usage of secure keys on the z/TPF system.
  - Includes program name, operation and cipher for each key used

```
ZDCOM SKEY DISPLAY SUMMARY
CSMP0097I 08.12.25 CPU-B SS-BSS  SSU-HPN  IS-01
DCOM0018I 08.12.25 THE SECURE KEY COMPLIANCE SUMMARY DISPLAY
  PROGRAM  OPERATION  KEY NAME   CIPHER
  -------  ---------  --------  ---------
    QCUS     ENCRYPT   CUSTINEK  AES256CBC
    QCUS     DECRYPT   CUSTINDK  AES256CBC
    QCRD     DECRYPT   CREDKYD   AES128CBC
    QLOY     ENCRYPT   LOYALEKY  AES256CBC
    QLOY     DECRYPT   LOYALDKY  AES256CBC
END OF DISPLAY
```



Christine
Security
Administrator

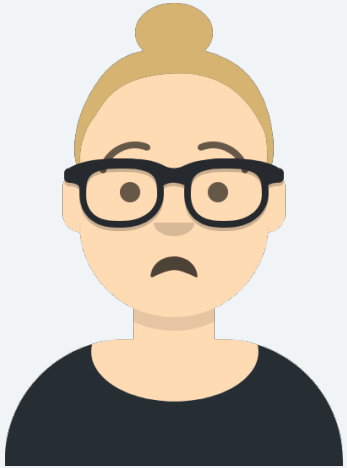
- The system maintains an inventory of key usage identifying which applications are using secure keys to perform encryption or decryption.

Note: z/TPFDF database use of secure keys is excluded from this display.

As-Is: Determining Usage of Ciphers on z/TPF

- Christine, a Security Administrator for the z/TPF system, is asked to ensure the AES128 cipher is no longer used on the z/TPF system. It is required to be AES256.
  - She needs to identify which applications are using AES128 to secure data in flight (network) or at rest (database).

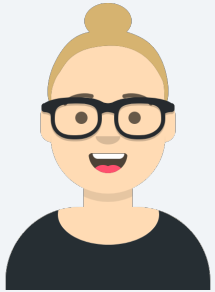
- Christine knows this will be difficult to determine. She needs to identify all the applications that are using TLS or encryption, and then must determine if the AES128 cipher is being used by any of the z/TPF applications.



Christine
Security
Administrator

To-Be: Determining Usage of Cipher Algorithms on z/TPF

- With a few commands, Christine can obtain
  - All the application usage of secure keys that use AES128
  - All the network ports that use AES128 type ciphers
  - All the z/TPFDF databases that use AES128



Displays Secure Key Usage

```
ZDCOM SKEY USAGE CIPHER-AES128CBC
DCOM0021I 08.35.32 THE SECURE KEY COMPLIANCE USAGE DISPLAY FOR
CIPHER AES128CBC
  PROGRAM   OPERATION   KEY NAME    CIPHER
  -------   ---------   --------  ---------
    QCRD     DECRYPT    CRDKYD     AES128CBC
```

Displays Network Usage

```
ZDCOM USAGE CIPHER-AES128-SHA
DCOM0006I 13.33.16 NETWORK COMPLIANCE USAGE DISPLAY FOR
CIPHER AES128-SHA

  PORT    MODEL     CIPHER USED
  -----  -------  -------------
  443    CLIENT         N
  443    SERVER         Y
  1000   SERVER         N
```

Displays z/TPFDF Database Usage

```
ZUDFM ENCRYPT USERSUM ENC-Y
FILE ID  DBNAME    ENCRYPTION  KEY NAME  CIPHER           DIGEST
-------  --------  ----------  --------  ---------------  ------
B211     CREDIT1   Y           DFCREDKY  AES256CBC        NONE
B221     FLIGHT1   Y           DFFLTKEY  AES128CBC        SHA256
```

Determining Key Usage in z/TPF Secure Keystore

- As symmetric keys are rotated, determining when it is safe to delete an old key can be problem.
  - Data might still exist in the database that is encrypted with that old key.
- The ZKEYS command was updated to display the date the key was last used for encryption or decryption.

```
ZKEYS DISPLAY LASTUSED

KEYS0056I 09.27.38 ZKEYS DISPLAY LASTUSED DISPLAY
ENC NAME DEC NAME ACT  ACT DATE   CIPHER  LAST USED
-------- -------- --- --------- --------- ---------
CUSTENC  CUSTD1    N  07JAN2021 AES256CBC 14FEB2022
CUSTEND  CUSTD2    N  09JAN2022 AES256CBC 15MAR2023
CUSTEND  CUSTD4    Y  06JAN2023 AES256CBC 25APR2023
PIDATA   PI_DEC1   N  12JUN2020 AES256CBC 12DEC2020
PIDATAD  PI_DEC2   Y  15OCT2022 AES256CBC 24APR2023
END OF DISPLAY
```

You can easily determine the last time a key was used to ‘help’ identify keys that can be deleted.

Gathering a Full Cryptographic Inventory on z/TPF

- In 2021, the z/TPF lab delivered z/TPF cryptographic inventory for networking.
- In 2022, the z/TPF lab delivered z/TPF cryptographic inventory for secure key usage.
- Documentation is now available with information about how to scan your source code for the remaining cryptographic operations.

Static Analysis of Cryptographic Inventory Example

Crypto.txt (zLinux based file)

```
tpf_RSA_sign
tpf_RSA_verify
tpf_RSA_encrypt_data
tpf_RSA_decrypt_data
tpf_cryptc
 CRYPC \+FUNC=
tpf_random
tpf_SHA1
tpf_SHA256
tpf_SHA512
tpf_encrypt_data
tpf_decrypt_data
 KLMD \+R[0-9]
 KIMD \+R[0-9]
 KM[ACFORT]* \+R[0-9]
```

Output of grep would exist in out.txt.

For more information, see "**z/TPF cryptographic function inventory code scanning**" in IBM Documentation.

https://www.ibm.com/docs/en/ztpf/latest?topic=inventory-ztpf-cryptographic-function-code-scanning

Value Statement

This enhancement makes it easier for you to gather z/TPF security compliance information for z/TPFDF databases and secure key usage:

- You can determine which z/TPFDF databases are encrypted and with which ciphers.
- You can determine which secure keys are being used on the z/TPF system
- You can scan their source code to compile a complete Cryptographic Inventory being used by the z/TPF system.

Delivered with **APARs PJ46863/PH48201** (Dec 2022)

Agenda

- z/TPF TCP/IP network override support
- Elliptic curve cryptography (ECC) support for secure network connections
- z/TPF cryptographic inventory
- **z/TPF TCP/IP performance enhancements**
- Support for latest cryptographic hardware on IBM z16

TCP/IP and OSA Performance Improvements

- The following TCP/IP and OSA-Express performance improvements were made on the z/TPF system.
  - More efficient blocking of output packets being sent to the OSA-Express card.
  - Reduction of number of times OSA-Express polling called.
  - Reduced pathlength for streaming outbound sockets.
  - More efficient processing of the OSA-Express input queue within the z/TPF task dispatcher.

- Performance improvements will benefit all environments (shared PR/SM, dedicated PR/SM)
- Nothing needed to enable the improvements, just install the enhancement APAR.

TCP/IP and OSA Performance Improvements – Performance Highlights

- Up to a 40% reduction in CPU utilization to process outbound streaming socket workload
- Up to a 49% reduction in calls to OSA Polling.
- Up to a 23% improvement in mixed transactional and outbound streaming workloads

Delivered with **APAR PJ46932** (Feb 2023)

Agenda

- z/TPF TCP/IP network override support
- Elliptic curve cryptography (ECC) support for secure network connections
- z/TPF cryptographic inventory
- z/TPF TCP/IP performance enhancements
- **Support for latest cryptographic hardware on IBM z16**

Crypto Express 8S

- The Crypto Express hardware is used by z/TPF to perform RSA operations.
    - For example, during SSL session startup.

- The IBM z16 processor delivered the next generation of Crypto Express card, Crypto Express 8S.

- The z/TPF system now supports the latest Crypto Express card.
    - Used to perform 1024-bit and 2048-bit RSA key operations.

- Delivered with **APAR PJ46507** (Mar 2022)
    - This is the only APAR required to move to a z16 (from z15)

What’s Next?

Disclaimer

Any reference to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.

Upgrading Version of OpenSSL – OpenSSL 3.0

- In September of 2023, the current z/TPF version (OpenSSL 1.1.1) is going out of support.
- An effort is underway to port the latest OpenSSL version 3.0.x.
- As part of this effort, will be prototyping the enablement of the latest TLS version TLS 1.3.
  - TLS 1.3 support will be delivered as a follow-on to OpenSSL 3.0.
- OpenSSL 3.0 is currently targeted for delivery in the summer of 2023

Thank you