

z/OS 3.1 IBM Education Assistant

Solution Name: GIMZIP Software Package Signing

Solution Element(s): SMP/E and z/OSMF Software Management

July 2023



Agenda

- Trademarks
- Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Upgrade & Coexistence Considerations
- Installation & Configuration
- Summary
- Appendix

Trademarks

- See url <http://www.ibm.com/legal/copytrade.shtml> for a list of trademarks.
- Additional Trademarks:
 - None

Objectives

- Increase confidence in the authenticity (who produced it?) and the integrity (has it changed in transit?) of **software delivery packages**.
- Who (Audience)
 - z/OS platform software installers
- What (Solution)
 - SMP/E will digitally sign GIMZIP packages
 - SMP/E will verify the signature for signed GIMZIP packages
 - z/OSMF Software Management will digitally sign Portable Software Instances
 - z/OSMF Software Management will verify the signature for signed Portable Software Instances
- Wow (Benefit / Value, Need Addressed)
 - Increased trust in the software you install

Overview

- SMP/E (GIMZIP utility) creates packages of portable files from z/OS data sets containing SMP/E consumables and SMP/E installed software.
 - z/OSMF Portable Software Instances (ServerPac)
 - CBPDO
 - Shopz PTF orders
 - SMP/E RECEIVE ORDER PTF and HOLDDATA orders
- Currently GIMZIP calculates a SHA-1 hash for each file and for the package.
- SMP/E and z/OSMF Software Management will be extended to:
 - Calculate and verify SHA-256 hash for each file.
 - Digitally sign the package.
 - Verify the signature of a signed package.
- A signed GIMZIP package will be compatible with existing SMP/E acquisition processing, thus signature verification is optional.

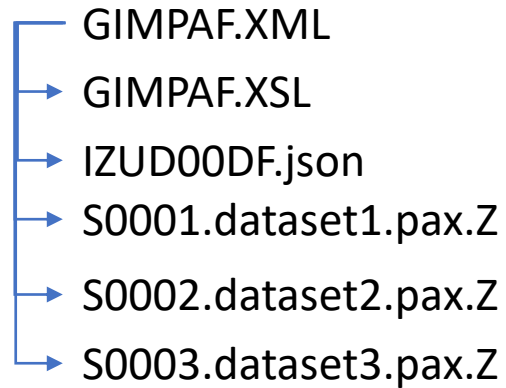
Overview...

- GIMZIP package signing is implemented using public/private key technology
 - A private key is used to calculate digital signatures.
 - The corresponding public key is used to verify the signatures.
 - The key pair is associated with an X.509 certificate.
 - The certificate, and associated private key, are used for signing.
 - The certificate, and associated public key, are used for signature verification.
- The signing certificate is issued by a well known and trusted certificate authority
 - The certificate authority establishes the authenticity of the package signer (is the signer who they say they are?).
 - If the certificate authority is trusted, so then a signing certificate issued by that certificate authority can also be trusted.
 - Therefore, to verify the signature of signed GIMZIP packages you must tell SMP/E which trusted certificate authorities may be used to validate the signing certificate and determine if the signer of the package is trusted.
- The signing certificate for the GIMZIP packages produced for IBM's z/OS product and service offerings is issued by the IBM z/OS certificate authority, **STG Code Signing Certificate Authority - G2**.
 - This CA certificate is built-in to RACF and other security managers.

Overview... GIMZIP Package Content, Unsigned

Existing GIMZIP package content:

/PackageDirectory



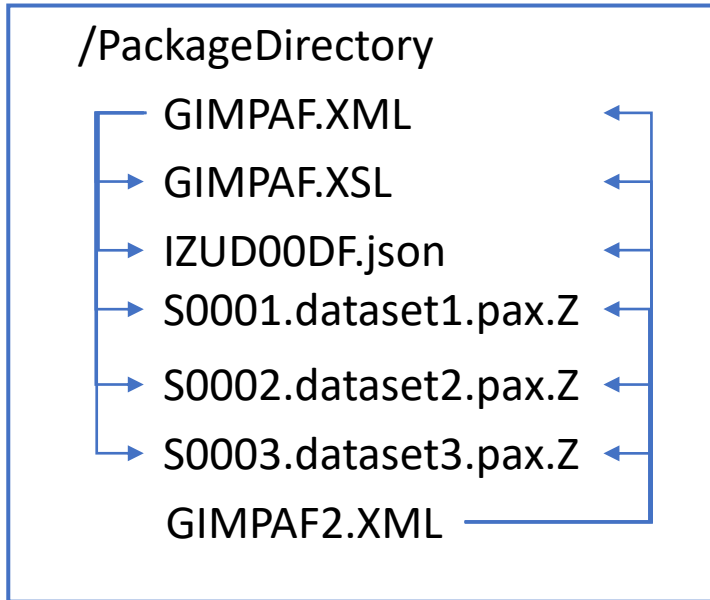
A diagram showing the directory structure of a GIMZIP package. It features a vertical line on the left with horizontal arrows pointing to the right, indicating a list of files. The files are: GIMPAF.XML, GIMPAF.XSL, IZUD00DF.json, S0001.dataset1.pax.Z, S0002.dataset2.pax.Z, and S0003.dataset3.pax.Z.

- GIMPAF.XML
- GIMPAF.XSL
- IZUD00DF.json
- S0001.dataset1.pax.Z
- S0002.dataset2.pax.Z
- S0003.dataset3.pax.Z

- GIMPAF.XML file:
 - Identifies all files in the package.
 - Contains SHA-1 hash for each file.
 - Contains SHA-1 hash for the package.

Overview... GIMZIP Package Content, Signed

Signed GIMZIP package content:



- **GIMPAF.XML file (Unchanged):**
 - Identifies all files in the package. *
 - Contains SHA-1 hash for each file.
 - Contains SHA-1 hash for the package.
- **GIMPAF2.XML file:**
 - Identifies all files in the package.
 - Contains SHA-256 hash for each file.
 - Contains SHA256withRSA signature for the package.
 - Contains certification path for the signing certificate, used for signature validation.

Overview... GIMZIP Package Acquisition

Without signature verification

No changes to input for GIMGTPKG
or RECEIVE FROMNET:

```
<SERVER...  
file="/orderdir/GIMPAF.XML"  
hash="3A1B4C2D..." >  
</SERVER>
```

/PackageDirectory

- GIMPAF.XML
- GIMPAF.XSL
- IZUD00DF.json
- S0001.dataset1.pax.Z
- S0002.dataset2.pax.Z
- S0003.dataset3.pax.Z
- GIMPAF2.XML

/PackageDirectory

- GIMPAF.XML
- GIMPAF.XSL
- IZUD00DF.json
- S0001.dataset1.pax.Z
- S0002.dataset2.pax.Z
- S0003.dataset3.pax.Z

Download

Looks just like an existing
GIMZIP package!

Overview... GIMZIP Package Acquisition

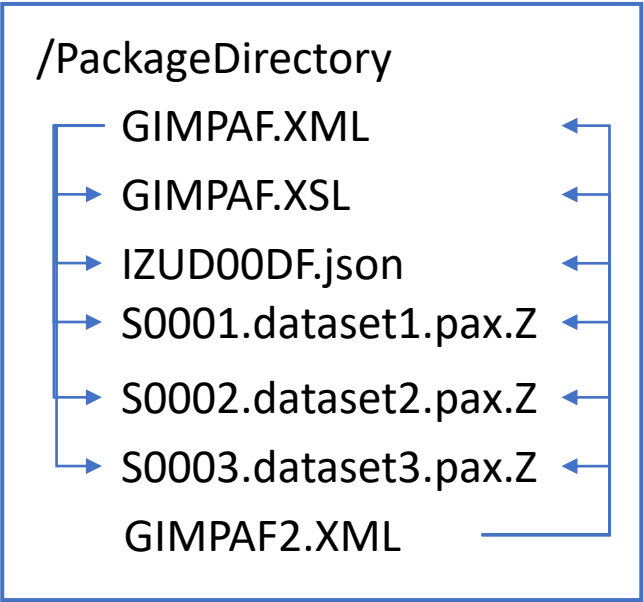
Without signature verification

No changes to input for GIMGTPKG or RECEIVE FROMNET:

```
<SERVER...  
file="/orderdir/GIMPAF.XML"  
hash="3A1B4C2D..." >  
</SERVER>
```



Looks just like an existing GIMZIP package!

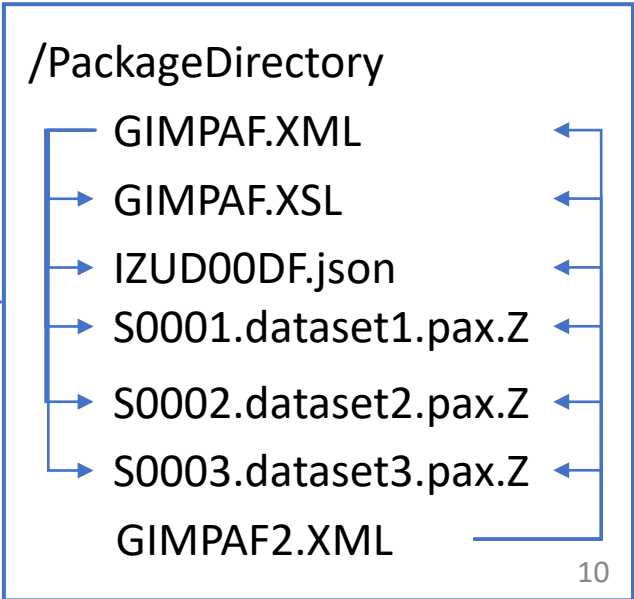


Download

With signature verification

Input for program GIMGTPKG or RECEIVE FROMNET:

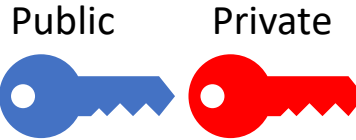
```
<SERVER...  
file="/orderdir/GIMPAF.XML"  
hash="3A1B4C2D..." >  
</SERVER>  
  
<CLIENT...  
signaturekeyring="IBM.gimzip.verify">  
</CLIENT>
```



Overview... IBM One-Time Setup

1. **Generate** a certificate with public/private key pair.
2. **Sign** the certificate with existing IBM z code signing root.
3. **Store** the signed certificate, its certification path, and private key in RACF on IBM Software Manufacturing production z/OS.

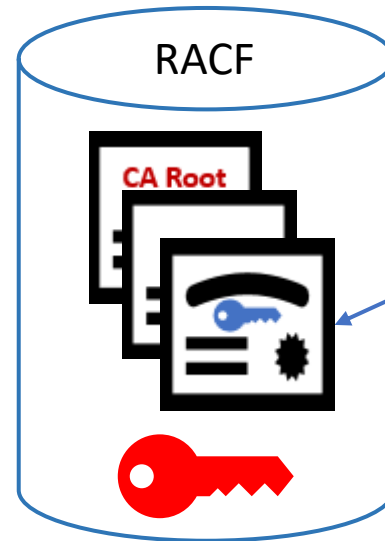
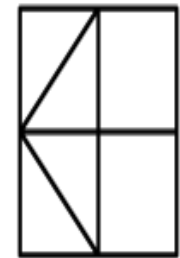
Public / Private Key Pair



Certificate



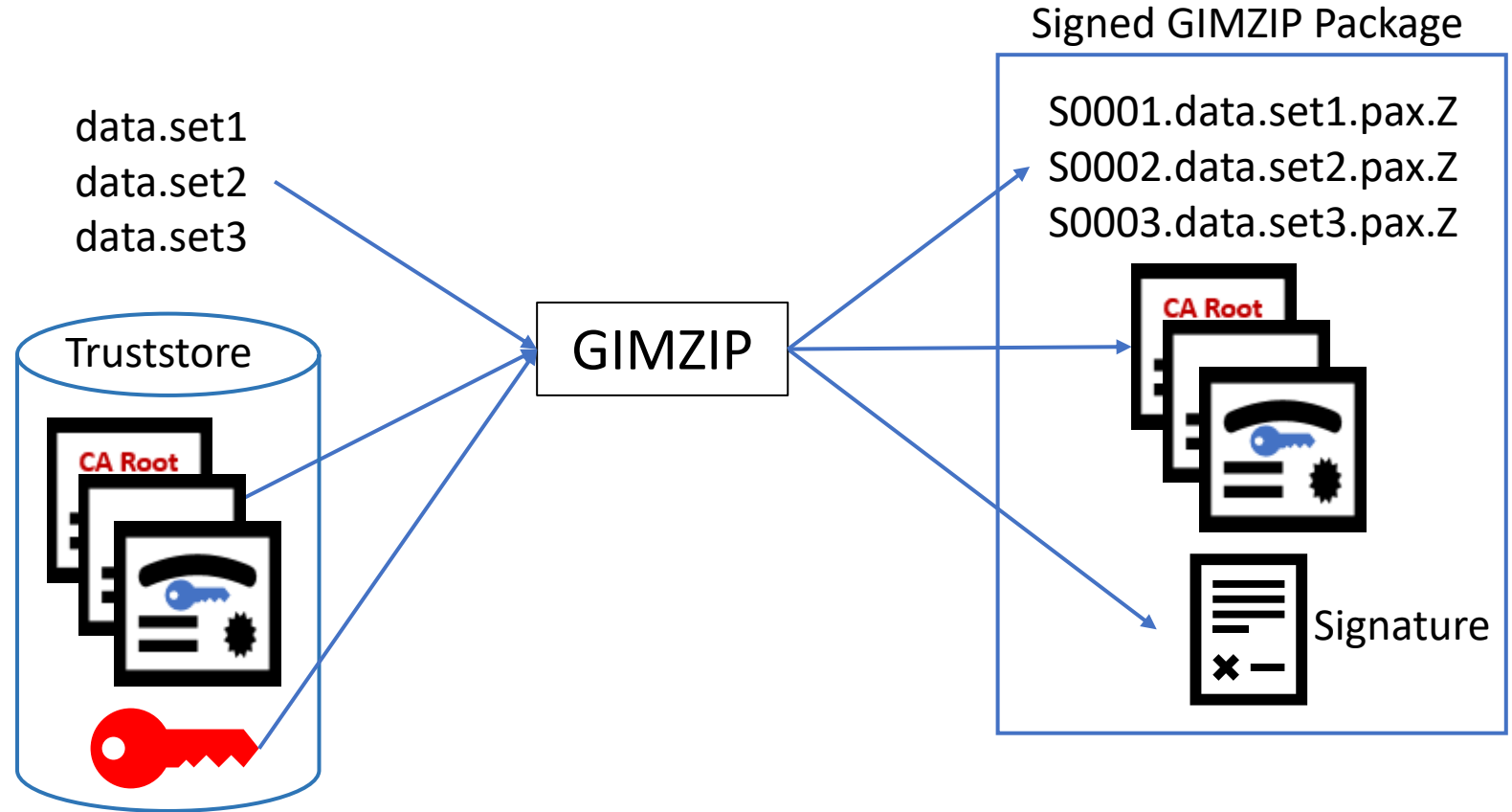
z/OS Development Root Certificate:
"STG Code Signing CA - G2"
Supplied with z/OS RACF



https://www.ibm.com/docs/en/zos/2.5.0?topic=guide-listings-racf-supplied-certificates#supcalist_ibmstg2

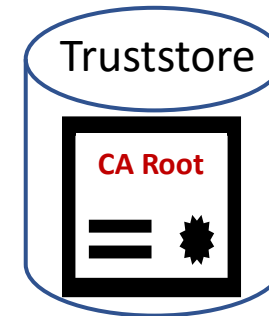
Overview... GIMZIP Signing Process

- 1. Discover and Validate** the certification path.
- 2. Create** archive files for each data set.
- 3. Write** the certification path to the package.
- 4. Sign** the package using the private key.



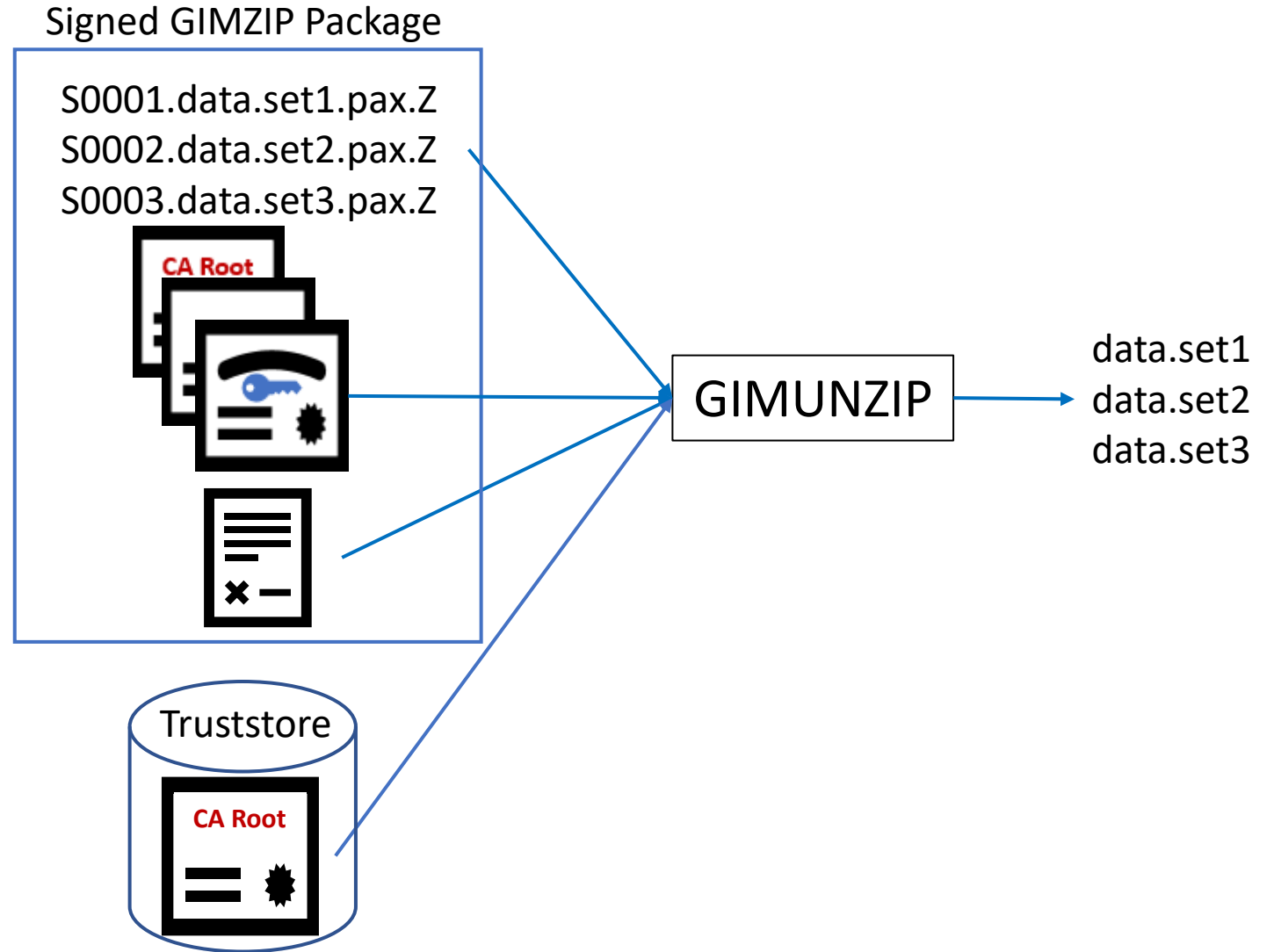
Overview... Consumer One-Time Setup

1. **Connect** the IBM CA root certificate to a keyring in your security manager db.



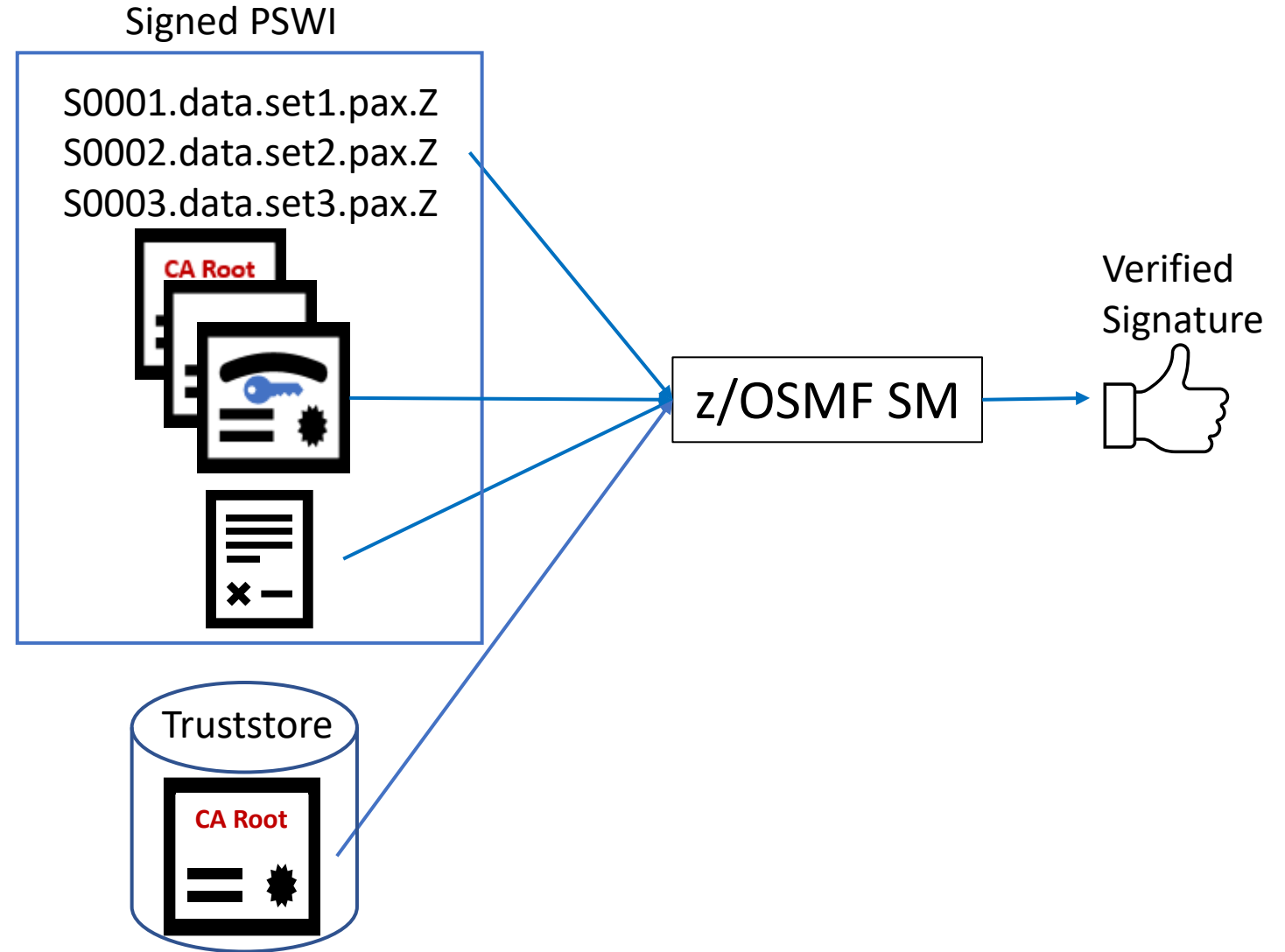
Overview... GIMUNZIP Signature Verify Process

1. **Validate** certification path using the CA root certificate in the local truststore.
2. **Verify** package signature using the public key.
3. **Create** data sets from archive files.



Overview... z/OSMF SM Signature Verify Process

1. **Validate** certification path using the CA root certificate in the local truststore.
2. **Verify** package signature using the public key.
3. **Persist** verified signer information.



Usage & Invocation... SMP/E

RECEIVE ORDER, RECEIVE FROMNET, GIMGTPKG

Verifying a package signature is optional.

1. A provider can sign packages, but **continue supplying <SERVER>** to consumers **unchanged** (file = GIMPAF.XML and SHA-1 hash)
2. Consumers can continue to download packages with existing levels of SMP/E
 - Signatures will not be verified

```
//SMPSRVR DD *
<SERVER
  host="download.server.com"
  user="S679p074"
  pw="k09944D4604223r">
  <PACKAGE
    file="/2022102123341/PROD/GIMPAF.XML"
    hash="3A14791D9F3DAA8D3DB25499538EEFBCAB5467F8"
    id="21October2022">
  </PACKAGE>
</SERVER>
/*
//SMPCLNT DD *
<CLIENT
  javahome="/usr/lpp/java/J8.0_64"
  downloadmethod="https"
  downloadkeyring="*AUTH*/*"
>
</CLIENT>
/*
```


Usage & Invocation... SMP/E

RECEIVE ORDER, RECEIVE FROMNET, GIMGTPKG

Verifying a package signature is optional.

1. If signature verification is desired, specify new attribute in <CLIENT> to identify SAF keyring name for root certificate
2. If the GIMPAF2.XML file resides on the server, it is downloaded and the signature verified
3. If the GIMPAF2.XML file does not reside on the server, processing will continue for the unsigned package

```
//SMPSRVR DD *
<SERVER
  host="download.server.com"
  user="S679p074"
  pw="k09944D4604223r">
  <PACKAGE
    file="/2022102123341/PROD/GIMPAF.XML"
    hash="3A14791D9F3DAA8D3DB25499538EEFBCAB5467F8"
    id="21October2022">
  </PACKAGE>
</SERVER>
/*
//SMPCLNT DD *
<CLIENT
  javahome="/usr/lpp/java/J8.0_64"
  downloadmethod="https"
  downloadkeyring="*AUTH*/*"
  signaturekeyring="IBM.package.sig.verification"
>
</CLIENT>
/*
```

Usage & Invocation... SMP/E

GIMUNZIP

Verifying package signature is optional.

1. If signature verification is desired, specify new EXEC parameter and attribute in <CLIENT> to identify SAF keyring name for root certificate
2. SMP/E and GIMUNZIP write a signature information message

```
//UNZIP EXEC PGM=GIMUNZIP,PARM='VERIFYSIG=YES'  
...  
//SMPCLNT DD *  
<CLIENT  
  javahome="/usr/lpp/java/J8.0_64"  
  signaturekeyring="gimunzip.verify.keyring"  
>  
</CLIENT>  
/*
```

```
GIM69270I  SIGNATURE VALIDATION FOR FILE "/u/ibmusr6/smpnts/test/GIMPAF2.XML"  
            WAS SUCCESSFUL. THE GIMZIP PACKAGE WAS SIGNED BY A CERTIFICATE WITH  
            SUBJECT NAME "CN=Kurts Package Signing Cert, O=IBM System Z, C=US",  
            SERIAL NUMBER "1" AND SHA256 FINGERPRINT  
            "4aa0fc6708314ca95fc2699bad116158298808c089f43e1ed4600eb4170916f4".  
            THE SIGNING CERTIFICATE WAS ISSUED BY "CN=Kurts Root CA, O=IBM  
            System Z, C=US".
```

Usage & Invocation... z/OSMF SM

Portable Software Instance Add Action

New option on all 3 Portable Software Instance **Add** actions to verify the signature for a portable software instance

1. From z/OS System
2. From Local Workstation
3. From Download Server

- Specify the signature verification SAF keyring

If the option is chosen the signature is verified for the portable software instance

The screenshot displays the 'Software Management' console interface. The breadcrumb trail indicates the path: 'Software Management > Portable Software Instances > Add Portable Software Instance'. The main heading is 'Add Portable Software Instance'. The form contains the following fields and controls:

- * System:** A dropdown menu showing 'pev171' and a 'Select...' button.
- * File location (UNIX file):** A dropdown menu showing '/u/zosmft6/swi4demo'.
- * Signature verification keyring:** A dropdown menu showing 'ibmusr6/gimunzip.verify.keyring' with an information icon.
- Verify the signature of the portable software instance:** A checkbox that is checked.
- Retrieve:** A button at the bottom of the form.

Usage & Invocation... z/OSMF SM

Portable Software Instance Add Action

- If the signature is verified, then the signer information is displayed

The screenshot shows the 'Software Management' console window. The breadcrumb trail is 'Software Management > Portable Software Instances > Add Portable Software Instance'. The page title is 'Add Portable Software Instance'. Below the title, there is a messages bar showing 1 information message. The message content states: 'The digital signature for the portable software instance has been successfully verified. The portable software instance was signed by a certificate with subject name "CN=Kurts Package Signing Cert, O=IBM System Z, C=US", serial number "1", and SHA256 fingerprint "4aa0fc6708314ca95fc2699bad116158298808c089f43e1ed4600eb4170916f4". The signing certificate was issued by "CN=Kurts Root CA, O=IBM System Z, C=US".' The message includes a timestamp of 'Oct 24, 2022, 4:14:56 PM' and a close button. Below the message, the 'System' dropdown is set to 'pev171' with a 'Select...' button. The 'File location (UNIX file):' dropdown is set to '/u/zosmf6/swi4demo'. The 'Verify the signature of the portable software instance.' checkbox is checked. The 'Signature verification keyring:' dropdown is set to 'ibmusr6/gimunzip.verify.keyring'. A 'Retrieve' button is at the bottom left.

Software Management

Software Management > Portable Software Instances > Add Portable Software Instance

Help

Add Portable Software Instance

Messages 0 0 1 Close All

The digital signature for the portable software instance has been successfully verified. The portable software instance was signed by a certificate with subject name "CN=Kurts Package Signing Cert, O=IBM System Z, C=US", serial number "1", and SHA256 fingerprint "4aa0fc6708314ca95fc2699bad116158298808c089f43e1ed4600eb4170916f4". The signing certificate was issued by "CN=Kurts Root CA, O=IBM System Z, C=US".

System: pev171 Select...

File location (UNIX file): /u/zosmf6/swi4demo

Verify the signature of the portable software instance. ☒

Signature verification keyring: ibmusr6/gimunzip.verify.keyring

Retrieve

Usage & Invocation... z/OSMF SM

Portable Software Instance View Action

- If the signature is verified, then the signer information is persisted and displayed on the Portable Software Instance View page

The screenshot shows a web interface for 'Software Management'. The breadcrumb trail is 'Software Management > Portable Software Instances > View Portable Software Instance'. The page title is 'View swi4demo'. There are three tabs: 'General', 'Products', and 'Digital Signature', with the last one being active. The content area shows the result of a signature verification: 'Was the signature verified? Yes'. Below this, it states 'Signature verification keyring: ibmusr6/gimunzip.verify.keyring'. A section titled 'Signing Certificate Details' contains a table with the following data:

Subject Name	Serial Number	Fingerprint	Issuer
CN=Kurts Package Signing Cert, O=IBM System Z, C=US	1	4aa0fc6708314ca95fc2699bad116158298808c089f43e1ed4600eb4170916f4	CN=Kurts Root CA, O=IBM System Z, C=US

Interactions & Dependencies

- Software Dependencies
 - None.
- Hardware Dependencies
 - None.
- Exploiters
 - IBM plans to exploit GIMZIP package signing for all z/OS software deliverables:
 - z/OSMF Portable Software Instances (ServerPac)
 - CBPDO
 - Shopz PTF orders (internet delivery only, physical DVD orders will not be signed)
 - SMP/E RECEIVE ORDER PTF and HOLDDATA orders
 - PTF orders are planned for a later date.

Upgrade & Coexistence Considerations

- To exploit this solution, all systems in the sysplex must be at the new z/OS level:
 - **No!**
- Toleration/coexistence APARs/PTFs:
 - z/OSMF – APAR PH49385
 - HSMA244 = UI83645
 - HSMA254 = UI83644

Installation & Configuration

- No changes to installation.
- Configuration:
 - Create a keyring containing the IBM CA root:
 - Detailed instructions: <https://www.ibm.com/docs/en/zos/2.5.0?topic=guide-preparing-verify-signatures-gimzip-packages>
 - Summary:

```
RACDCERT ID(userid) ADDRING(IBM.package.signature.verification)
RACDCERT ID(userid) CONNECT(LABEL('STG Code Signing CA - G2') +
RING(IBM.package.signature.verification) USAGE(CERTAUTH) )
```

- When installing z/OS 3.1, specify your signature verification keyring on the z/OSMF Portable Software Instance Add action (described on previous slides)

Summary

- Increasing confidence in the authenticity (who produced it?) and the integrity (has it changed in transit?) of **software delivery packages**, so you can trust the software you install.
- Updates to sign packages:
 - SMP/E GIMZIP service routine
 - z/OSMF Software Management, Software Instance, Export action
- Updates to verify signatures for signed GIMZIP packages:
 - SMP/E GIMUNZIP, GIMGTPKG, RECEIVE FROMNETWORK, RECEIVE ORDER
 - z/OSMF Software Management, Portable Software Instance, Add action

Appendix

- SMP/E User's Guide, "Preparing to verify signatures for GIMZIP packages"

<https://www.ibm.com/docs/en/zos/2.5.0?topic=guide-preparing-verify-signatures-gimzip-packages>

- SMP/E Commands, "RECEIVE Command, <CLIENT> XML"

<https://www.ibm.com/docs/en/zos/2.5.0?topic=processing-content-client-data-set>