Keymaker

PKWARE Key Maker Overview

Organizations that rely on files encrypted with OpenPGP need a fast, reliable way to encrypt and decrypt OpenPGP files. They also need a method of ensuring the people who handle OpenPGP files can easily create and open these files. OpenPGP users identify themselves, and develop trust through public and private keys.

PKWARE provides SecureZIP to encrypt and decrypt strongly-encrypted files using passphrases, X.509 certificates and OpenPGP keys. SecureZIP Server eBusiness Edition includes PKWARE Key Maker to allow you to create and manage OpenPGP keys. This guide will walk you through the basics of using PKWARE Key Maker. Key Maker also features a graphical interface that allows you to work with OpenPGP keys in a familiar point-and-click manner. This help system offers assistance in carrying out Key Maker tasks.

For more information about SecureZIP, see http://www.pkware.com/software/securezip/

Use of PKWARE Key Maker is covered under the terms and conditions of your SecureZIP license agreement.

Introduction to Open PGP

Some organizations use encryption tools based on the OpenPGP standard, rather than X.509. OpenPGP uses the same basic Public Key Infrastructure principles for exchanging encrypted files, but uses a decentralized “Web of Trust” method of authenticating signatures.

SecureZIP extracts and decrypts files that comply with the OpenPGP specification defined by the Internet Engineering Task Force RFC 4880. SecureZIP can also create OpenPGP-compliant files and sign files with OpenPGP keys.

OpenPGP keys are typically created by individuals, and authenticated by other individuals. In the real world, you have friends who can vouch that you are who you say you are. If you walk into a room full of strangers, your friend can introduce you to the people he knows. Since you trust that your friend is correctly identifying his friends and acquaintances, your trust extends to his friends too.

When you translate the above experience to the electronic, OpenPGP world, it works this way: You create an OpenPGP key to identify yourself. When a friend comes to visit, display the key. The friend can now sign your key (often called “key signing”) and certify that this key represents you. Now everyone who trusts the person who signed your key can also trust that your key is authentic. A Web of Trust is developed as more people authenticate each key. Everyone in the Web of Trust can also exchange messages in the OpenPGP format.

In order to use OpenPGP keys with SecureZIP, they must first be generated and stored in an OpenPGP compliant key repository. Typically, this repository is a keyring file. OpenPGP public keys are stored in a public keyring file. While not required by the OpenPGP standard, or by PKWARE Key Maker, public keyring files usually have a file extension of .pkr. OpenPGP secret keys are stored in a secret keyring file. Secret keyring files usually have a file extension of .skr. Other file extensions may be used for keyring files. PKWARE recommends using the .pkr and .skr file extensions respectively when referencing public and secret keyring files, but other keyring file extensions can be used with this program. The PKWARE Key Maker program provides a means of creating OpenPGP keys and keyring files for use with SecureZIP.

> Where your keyring is stored may depend on the software used to create the keyring. Most OpenPGP tools for Windows (including PKWARE Key Maker) store the keyring file by default in C:\users\<username>\My Documents\pgp. GnuPG stores the keyring file in C:\users\<username>\APPDATA\Roaming\gnupg. On UNIX and Linux systems, keyrings are typically stored in /home/<username>/ .pgp or /home/<username>/.gnupg directory.
>
> PKWARE Key Maker by default searches these locations for existing keyrings.

Use the Key Maker Settings dialog box to define your existing public and private keyrings if they are not stored in either of the default folders.

General Operations

Generating OpenPGP Keys

To generate a new OpenPGP public/private key pair:

1. Click **New Key** from the button bar (or go to the **Keys** menu and select **Create New Key Pair**).
2. Define the required characteristics of this key:
    a. **Key Type**: Determines the type of key to create. Possible values for OpenPGP are RSA (default) and DSA.
    b. **Key Size**: Key length when generating new keys. For RSA, possible values are 1024, 2048 (default) and 4096. DSA will use the same values and defaults as RSA but they will apply to the El Gamal encryption key – not the DSA signing key.
    c. **User ID**: OpenPGP userid to be used in OpenPGP key creation or for locating an OpenPGP key in a keyring. This value can contain a name, email address and comment; such as: Tom tom@example.com.
    d. **Passphrase**: Output passphrase, used to protect generated private key.
3. (Optional) Set an expiration date for this key.
4. Click **OK** to create key pair.

Signing OpenPGP Keys

Establish trust relationships with other OpenPGP keys by signing these keys.

1. Select the key you want to sign from the list.
2. Click **Sign** from the button bar (or from the Keys menu). The Sign Key dialog box appears.
3. If necessary, you can use the drop-down menu in the Key field to change the selected key.Identify the OpenPGP secret key to sign the key with in the **Sign with** line.
4. Enter the passphrase for the key you're signing with.
    a. (Optional) Limit the extent of the trust you're giving with this signature:
        i. **Local**: You're signing this key only for others in the current keyring
        ii. **Exportable**: The signature can be included in other keyrings (including public repositories)
5. **Expires on**: Check the box, and assign an expiration date to your signature.

Click **OK** to sign the key.

Remove a Key from a Keyring

To remove an OpenPGP key from a keyring:

1. Select the key you want to remove from the list.
2. Click **Remove** from the button bar (or from the Keys menu).
3. Confirm that you want to remove the selected key. Click **Cancel** to keep the key.

CAUTION: Only remove keys that are not associated with any OpenPGP file or message.

Key Information

When you click to select a key from your keyring, Key Maker displays the following information:

General Tab

Field	Description
Primary User ID	The userid value can contain a name, email address and comment; for example: Tom <tom@example.com>
KeyID	Used to identify a particular OpenPGP key by its unique key ID. The short KeyID (displayed first) are the last eight characters of the Fingerprint (listed below), and the long KeyID (in parentheses) are the last 16 characters of the Fingerprint
Type	Public or Key Pair (public and private)

Size	Number of bits in the key
Validity	Whether a key is valid, revoked, disabled, or expired
Trust	Assigns the level of scrutiny the person associated with this key gives before signing another key. When first created, the key's trust level is **Unknown. Other trust levels include Marginal**, **Complete** and **None.** The **Implicit** trust level should only be assigned to your own keys.
Created	Date the key was created
Expires	Date the key is no longer valid
Cipher	A list of encryption algorithms marked as "preferred" for people using the key. Keys made by Key Maker specify these algorithms (in order): AES-256, AES-192, AES-128, CAST5, and 3DES.
Fingerprint	The complete unique string of characters for this key.

User IDs tab

Field	Description
Name	Common name and email address associated with this key
Type	This field will always be **UserID**

Signatures tab

Field	Description
Type	Specifies the encryption algorithm used to sign the key. **DSA** keys can only sign. **RSA** keys are also used to encrypt.
Signed User ID	Identifies the key that's been signed. This value can contain a name, email address and a comment of the signee.
Signer Name	Name (and often the email address) of the signer.
Signer Key ID	The unique eight-character ID for the signer
Created	Date the signature was created
Expires	Expiration date of the signature, if any.

Subkeys tab

You can attach a subkey to any primary public/private key pair to use the same key pair to sign and encrypt files. If your sub key is compromised, you don't need to revoke your master key.

Field	Description
SubkeyID	The unique identifying ID for the subkey
Algorithm	Specifies the algorithm used to encrypt the subkey. **RSA, ElGamal**, or **DSA** (if this is an additional signing subkey)
Valid From	Date the subkey was created
Expiry Date	Date the subkey is no longer valid
Size	The length (in bits) of the subkey
Status	Whether the subkey is **expired** or **revoked**

Advanced Operations

Setting a Non-Default Keyring Location

Most OpenPGP tools for Windows (including PKWARE Key Maker) store the keyring file by default in C:\users\<username>\My Documents\pgp. GnuPG stores the keyring file in C:\users\<username>\APPDATA\Roaming\gnupg. On UNIX and Linux systems, keyrings are typically stored in /home/<username>/ .pgp or /home/<username>/.gnupg directory.

PKWARE Key Maker searches these locations for existing keyrings. If your keyring is not in one of these default locations, use the Key Maker Settings dialog box to identify the appropriate keyring.

1. Go to **Main > Settings**.
2. Type (or Browse to) the full path to the Public Keyring (including the *.pkr file).
3. Type (or Browse to) the full path to the Private Keyring (including the *.skr file).
4. Click **OK** to confirm the changes.

When you have made your changes, Key Maker will always place new generated keys in the defined keyring. It will also use the defined keyrings for other operations (such as Import and Export).

Add a UserID to a Key

You can add a second UserID to a key if you want separate identities for different uses (personal and business, for example). To do this:

1. Select the key you want to work with from the list.
2. Click Add User from the button bar (or from the Keys menu). The Add New User ID dialog box appears.
3. If necessary, you can use the drop-down menu in the Key to Edit field to change the selected key.
4. Type the new User ID (name and email address) in the User ID to Add box
5. Type the passphrase for the main key you are adding to.
6. Click **OK**.

Create a New Subkey

Most OpenPGP keys have at least one subkey. You can attach a subkey to any primary public/private key pair to use the same key pair to sign and encrypt files. If your subkey is compromised, you only need to revoke the subkey, not your master key.

To add a subkey:

1. Select the key you want to add the subkey to from the list.
2. Click **New Subkey** from the button bar (or from the Keys menu). The **Create Subkey** dialog box appears.
3. If necessary, you can use the drop-down menu in the Master Key field to change the selected key.
4. Use the drop-down menu to select the key size for the subkey. The default is 2048-bit, you can also choose the more secure 4096-bit. You may also select the less secure 1024-bit, but this is not recommended.
5. Add a passphrase to the subkey.
6. Optionally, you can specify the **Start Date** and **Expiry Date**.
7. Click **OK** to create the subkey.

Exporting Keys

Use this command to export keys and keyrings from one location to another. In the command line interface, you can use the Copy command for this operation. This command allows you to copy one or more public keys or a keyring to another public keyring, or copying of one ore more secret keys or keyring to another secret keyring.

To export keys:

1. Select the key you want to export from the list.
2. Click **Export** from the button bar (or select **Export** from the **Keys** menu). The Export Key dialog appears.
3. If necessary, you can use the drop-down menu in the **Key to Export** field to change the selected key.
4. Select the Export Format from the drop-down menu.
    a. **Complete**: (Default) Exports all attributes for this key.
    b. **Compatible**: Exports only attributes for this key supported by older OpenPGP versions.
5. Select from the Options:
    a. Export private key: By default, Key Maker will only export the public key for this key pair. Check this box to export the private key with the public key. DO NOT check this when exporting to a publirepository!
    b. **Armored file**: Use ASCII armor for OpenPGP output file.
6. Click **OK** to export.

Importing Keys

Use this command to import keys and keyrings from one location to another. In the command line interface, you can use the Copy command for this operation. This command allows copying of one or more public keys or a keyring to another public keyring, or copying of secret keys or keyring to another secret keyring.

To import a single key to the existing keyring:

1. Click **Import** from the button bar (or select **Import** from the **Keys** menu).
2. Browse to the location of the key to import.
    a. By default, Key Maker displays **All Keyring Files** in the Import window. Use the **Files of Type** drop-down menu to select just OpenPGP files (with the .pgp or .gpg extension), or Armored files (with the .asc extension). ASCII armor (Radix-64) is a character format that creates an ASCII character stream that could be used in transferring OpenPGP files through transport mechanisms that can only handle character data (for example, email body text).
3. Click **Open** to import the selected key.

Key Maker Command Reference

The Key Maker graphical interface lets you perform common and simple tasks with OpenPGP keys.

Key Maker on the command line (included in SecureZIP Server eBusiness Edition) has many more capabilities and options, but also does the basic tasks that the graphical interface handles. This table identifies the equivalent CLI commands.

Add a UserID to a Key

Task	CLI Command	GUI Equivalent
Generating OpenPGP Keys	*generate*	**Keys > Create New Key-pair** OR New Key
Add a UserID to a Key	edit	**Keys > Add New UserID** OR Add User
Signing OpenPGP Keys	*sign*	**Keys > Sign** OR Sign
Exporting Keys	*copy*	**Keys > Export** OR Export
Importing Keys	*copy*	**Keys > Import** OR Import
Remove a Key from a Keyring	*delete*	**Keys > Remove** OR Remove