

Sophos Firewall OS Secure Storage

Frequently Asked Questions

Background

Why have Sophos introduced this new feature?

SFOS needs to store sensitive information for a number of purposes, including:

- authenticating users, supporting a wide range of different protocols
- communicating with directory services, to ensure correct policies and group memberships
- access external services such as email servers and FTP servers
- enabling use of TLS and other PKI-enabled protocols to secure communications channels

A recent security review determined that we could significantly improve the protection of the information we store. Improving our secure storage means that if an attacker gains unauthorized access to your firewall, it is much harder for them to discover and extract sensitive information.

I've heard it's best practice to use a one-way hash to store passwords. Why not just do that?

One-way hash functions are perfect when the only reason for storing the password is to validate that the user has entered it correctly at log-in. The user enters their password, the password is hashed immediately with a one-way function and then the hashed values are compared. Even if an attacker gets hold of the stored hash value, they can't work out the original password.

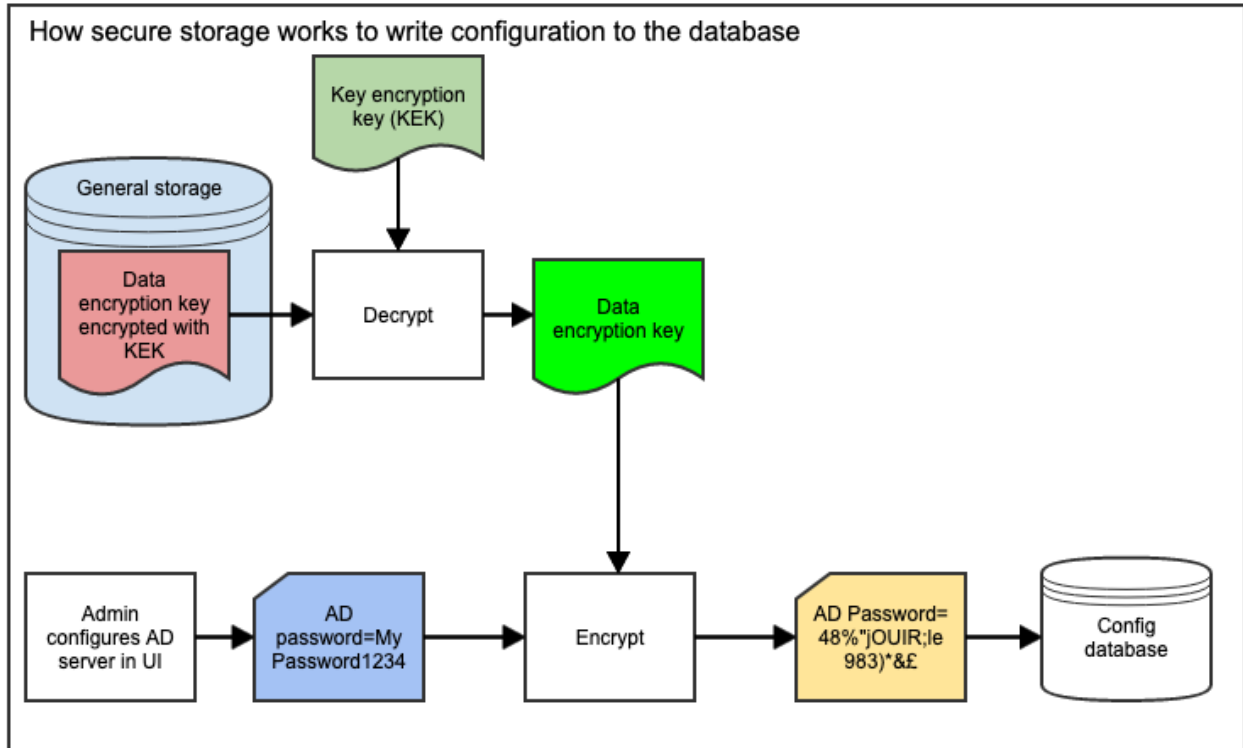
But much of the sensitive information stored on XG firewall needs to be used by the system in its original form. It's no good keeping a hashed version of the Admin password for your Active Directory server because when XG firewall tries to connect to the server it will need to provide the original password. We must store each passwords in a reversible form, which means encrypting it with a key and storing the key securely.

How does Secure Storage work?

As soon as your firewall upgrades to a firmware version that supports Secure Storage, SFOS creates two random keys – the Data Encryption Key (DEK) and the Key Encryption Key (KEK).

The DEK is used to encrypt all existing sensitive information on the device. It is also used to protect any sensitive information that is added or modified in the future.

The second key, the KEK, is used to encrypt the DEK before storing it on the device. The KEK itself is written to a secure location. The combination of keys allows SFOS to maximize security while making it easier to deal with future key changes.



When SFOS needs to use the sensitive information – for example, when the Firewall has to log in to an FTP server to write a scheduled backup file – the encrypted information is read from the database and the key is used at the last minute to decrypt it for use, after which the decrypted copy is deleted.

While the sensitive information remains on the XG Firewall system it's secure, and is only decrypted when needed. An attacker looking through the system will only see the information in its encrypted form and so they can't just bulk-extract passwords or keys at will.

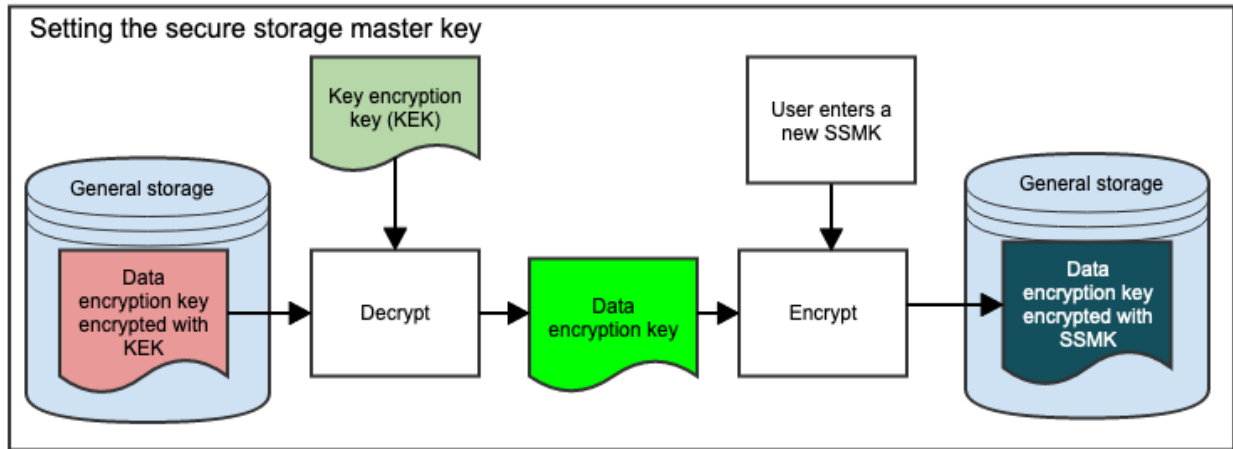
But sometimes it's necessary to move this sensitive information. Backups and configuration exports require the data to be moved or stored so that it can be used on a different system, without the same stored keys. That's where the secure storage master key comes in.

Secure Storage Master Key

What is the Secure Storage Master Key?

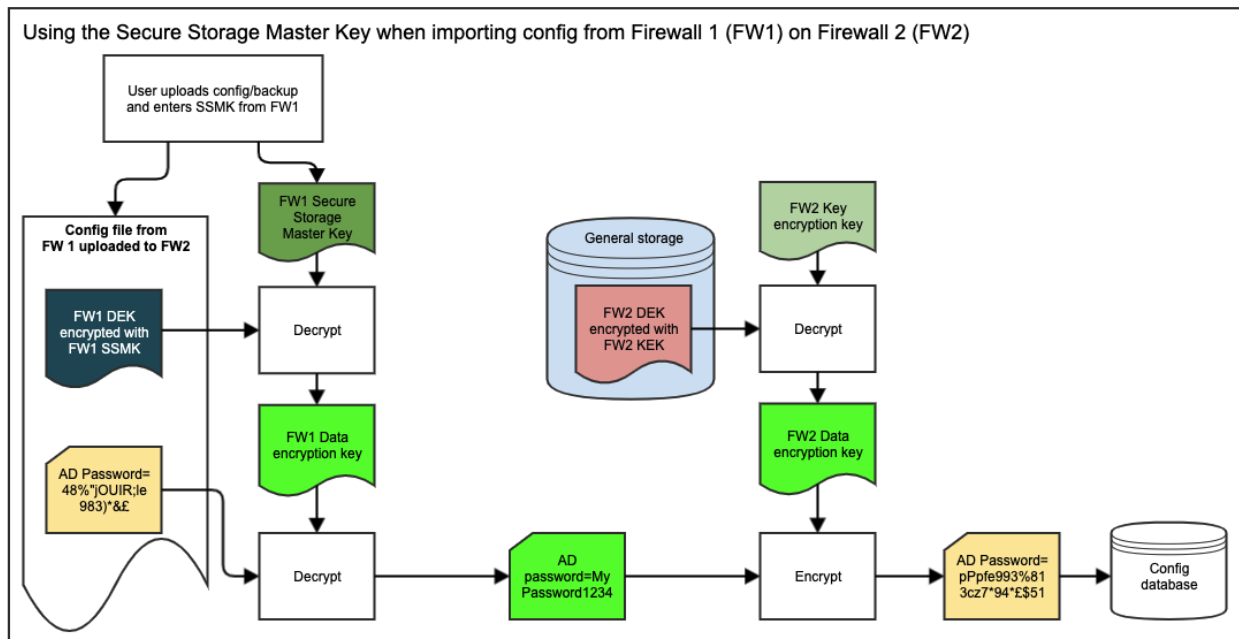
Setting the Secure Storage Master Key allows you to restore your backup or import a configuration on any other device.

When you set the Secure Storage Master Key, the firewall uses it to encrypt a new copy of the Data Encryption Key.



This encrypted copy is included in all the backups and config exports that are created from that point onwards.

When those backups or exports are used on another device in the future, you can unlock the Data Encryption Key with your Secure Storage Master Key and allow all the encrypted information to be restored.



How do I create the Secure Storage Master Key?

You can create the Secure Storage Master Key by logging in to your XG Firewall's admin UI as the default 'Admin' user.

When you've upgraded to a version of SFOS that supports secure storage, you will start to see notifications whenever you log in to the Admin UI. We suggest you read these notifications.

Create the secure storage master key ✕

What's the secure storage master key?
The secure storage master key provides extra protection in backups and imported configurations for account and password details stored on the firewall.

⚠️ Until you create the master key, scheduled backups will run, but won't benefit from the extra protection. For more details, go to [Secure storage master key](#).

When do you need to use the master key?
You need the master key when you restore a backup or import a configuration. This key is in addition to the backup encryption password.

Can you recover the master key?
If you lose the secure storage master key, you can't recover it. You can create a new one, but you can't restore backups or configurations created with the lost key.

Store the master key safely in a password management system or another secure location.

[Skip for now](#) [Create key](#)

If you log in as a locally-created Administrator account, you will see the notifications but will not be able to proceed to setting the key.


If you log in as the default 'Admin' account, you will be able to set the key.

Create the secure storage master key ✕

Before you create the master key, ensure that you can store the master key in a password management system or another secure location.

⚠️ If you lose the secure storage master key, you can't recover it.

Enter the secure storage master key



Key strength: **Strong**

Enter your key again to confirm

I have stored the master key in a password manager or another safe place

[Back](#) [Create key](#)

Complexity requirements:

- ✔️ Minimum 12 characters
- ✔️ An uppercase letter
- ✔️ A lowercase letter
- ✔️ A number (0-9)
- ✔️ A special character

What happens if I don't create a Secure Storage Master Key?

Until you've created the key, you won't be able to manually create or download backups. Scheduled backups will continue to happen as previously configured and will be fully restorable without the need for a Secure Storage Master Key.

If you export configuration before creating the Secure Storage Master Key, any sensitive data will be encrypted but there will be no way to import it. Other non-sensitive information will be imported but if it has dependencies on the sensitive information the import will be incomplete.

What if I forget my Secure Storage Master Key?

You will no longer be able to restore backups created with secure storage, or fully import previously-exported configurations.

If you think you may have lost or forgotten your Secure Storage Master Key, we recommend you reset it as soon as possible, to ensure that you start generating backups that you can restore in the event of a problem.

How do I reset my Secure Storage Master Key?

Connect to the command-line console on your Firewall. Select option 2 (System configuration) and then select option 5 (Reset secure storage master key). The prompts will take you through the process of creating a new master key.

Note that you will need to enter the password for the default 'Admin' account in order to reset the key.

Using backup with secure storage

I already have a backup password – why do I also need a secure storage master key?

Backup passwords encrypt the final backup file, an archive containing all the config and associated files to allow a complete restore. Their purpose is to provide protection for the data after it has been downloaded from the firewall. It is possible for any Admin user with the right login privileges to change the encryption password before downloading a backup file. This is convenient from a management point of view – it allows admins to change passwords at any time or use one-time passwords for temporary storage. But it also gives an attacker a way to potentially change the password to something they know and to recover the data inside the backup in that way.

The secure storage master key is harder to change, so it's harder for an attacker to manipulate the system and change the key to something they know. This additional layer of protection keeps sensitive information secret even if the backup password is known to an attacker.

So why is it impossible to restore a backup without the master key?

Much of the information in a backup is not encrypted with secure storage – network interface configurations, IP hosts, firewall rules and other policies. But there is a complex network of

relationships and dependencies between different types of objects. If a configuration is not restored completely, your firewall could end up in a state where it behaves in unexpected ways, because objects that rely on sensitive data are not there.

When we restore a configuration from a backup, we are taking a snapshot of the configuration database and recreating it in-place. We assume that the original database was fully consistent and operational and so restoring the snapshot will also be. Trying to restore a parts of that snapshot would make the restore process much more complex, slowing it down and making it more liable to failure.

In future, we may be able to provide a way to partially restore selected objects from backups when the secure storage master key is not known.

[What about backups stored in Sophos Central?](#)

Backups stored in Sophos Central behave like any other backup. Any new backups stored after the firewall has upgraded will contain information protected by secure storage. When restoring a backup that was stored in Central, you will need to enter the secure storage master key.

When downloading a backup from Sophos Central, you can change the backup password. This does not affect the secure storage master key.

[Using configuration import/export with secure storage](#)

This section relates to the use of the Import/Export feature in XG Firewall, which can be accessed at **Backup & firmware > Import export**.

[How does configuration export work with secure storage?](#)

When you export a configuration, your firewall generates an XML API document that can be used to re-create the same configuration on another device. When secure storage is in operation, sensitive data will be exported in its encrypted form.

To enable encrypted sensitive data to be imported on another device, you will need to provide the master key from the original device when you perform the import.

[I can't remember the master key – can I still import the configuration?](#)

SFOS will try to import as much of the configuration as possible, but it may leave your import in an incomplete state. Any configuration items with encrypted data will not be imported. Also, any configuration items that are dependent on those items will be skipped.

[Why can't SFOS import items with unmet dependencies?](#)

Unlike backup and restore, configuration import treats each separate item within the import as a separate operation using SFOS's XML Configuration API. For example, if you import a set of Firewall Rules, each rule will be created separately as if you were setting them up via the User Interface.

But if you try to create a Firewall rule that depends on another object that does not exist on the new device, the operation will fail. For example, if you firewall rule uses a DNS host object

called 'my-server.domain.com', that object must be created before the Firewall rule can be successfully imported.

When importing a configuration that is protected with secure storage, we will still attempt to recreate each item individually. Items that do not contain any protected information will successfully import. Items that do contain protected information will fail.

Where non-protected items have dependencies on failed items, those will also fail.

How do I know which items were successfully imported and which ones I'll need to re-create?

For now, you'll need to look through the configuration in the UI to determine which configuration items were skipped.

In a future release we'll be adding the ability to view a report summarizing the successes and failures during import.

Other questions

Do I need to do anything special when running in an HA configuration?

In short – No.

All encryption keys are shared and automatically synchronized between the two devices in an HA pair. That means the replication of confidential data between the devices is always encrypted and confidential data is always stored in an encrypted form, just like on a standalone device.

You only need to set the Secure Storage Master Key once for the HA pair, and the same key will apply for backups or exported configuration from either device.

If one of the HA devices fail and needs replacing, the key information will be automatically transferred to the new device during the initial synchronization.

Resetting the SSMK

Will I lose any data?

No. Resetting the SSMK does not impact any stored secrets or information. The Data Encryption Key does not change. The system just stores a new copy of the Data Encryption Key, protected with the new SSMK.

XML API

How does this affect the XML API?

Encrypted data is exported with a new property `hashform="mode1"`. Manually replacing these with unencrypted values would allow the XML API file to be imported without the SSMK.

```
<User transactionid="">
  <Username>simon</Username>
  <Name>Simon Says</Name>
  <Password hashform="model">$sfos$7$0$fdKXEL5H09EklwbnHmYC
  ~9oMWC-QApZ0axkw5csjJ0yE3S9tfKnxLg4oKZD5rRBA~</Password>
  <Description/>
```