Backup365 Security Overview

Passwords Storage *(Impersonation Credentials)*

It is not ideal to be storing passwords, but we need to pass email/password credentials into EWS in order to make a connection and sync the data so the password does need to be stored somehow. With this in mind, we have followed the best practices recommended by Amazon when it comes to password storage, we use a combination of KMS, S3, and DynamoDB to keep the passwords secure.

Account Impersonation

Our first step in secure storage of passwords is to minimise the number of passwords that need to be stored. Instead of storing a password for every account, we use an impersonation account for the EWS authentication and so only have to store one password (for the impersonation account) per organisation.

One of the main benefits of only storing one password per organisation is that the password can be very secure (and so hard to remember) and regularly changed without individual members of the organisation having to worry about such things. It also means that in a situation where the password becomes compromised, it can be easily reset without people having to worry about their personal passwords being leaked or having to change them.

Password Storage

The passwords that we do need to store are encrypted using AES-256 before being stored in the DelegateAccounts table in DynamoDB. Access to DynamoDB itself is restricted to only those with AWS login credentials (i.e. the dev team and the Manage Protect root account). The encryption is implemented using the PyCrypto library, which performs the AES encryption using CBC mode, and uses a randomly generated (using the library) "initialisation vector", which means that the same password encrypted multiple times will yield a different ciphertext.

Key Management

The key that is used in the encryption is managed using Amazon's Key Management Service (KMS). The key was generated using KMS and and stored encrypted in S3. The plaintext version of the key is never stored on disk. When the key is needed, a request is sent to KMS to decrypt the key. The plaintext key is sent back to the program, which uses it and then deletes it. This ensures that the plaintext key is only stored in memory for the shortest time possible.

Email Data

The email data is stored encrypted in S3 and access is protected by Amazon's security and is limited to only those with access to the AWS console. All of the email data stored in S3 is encrypted and has the same security afforded by KMS as the password storage.