sign-efi-sig-list

Outils de signature de variables

Produit un fichier de sortie avec un en-tête d'authentification pour mettre à jours une variable. Cette sortie peut être signée par les clés usuelles directement ou peut être splité pour une signature externe en utilisant les options -o et -i.

OPTIONS

-r Le certificat est rsa2048 au lieu de x509

-m Utilise un compteur monotonique au lieu d'un timestamp

-a Prépare la variable pour APPEND_WRITE au lieu d'un remplacement

-o Ne signe pas, mais sort un fichier du bundle exact à signer

-t timestamp utiliser le timestamp spécifié.

-i Prend une signature détachée au format PEM produit pas -o et complète la création de la mise à jours

-g guid Utilise le guid comme guid propriétaire de la signature

-c crt Le certificat de signature au format PEM

Exemples

Pour signer une simple mise à jours dans db qui a été préparé comme liste de signature EFI dans DB.esl et sort le résultat avec l'en-tête d'authentification dans DB.auth

sign-efi-sig-list -a -c KEK.crt -k KEK.key db DB.esl DB.auth

Pour faire une signature détachée :

sign-efi-sig-list -a -t 'Jul 21 09 :39 :37 BST 2012' -o db DB.esl DB.forsig

Signer le fichier DB.forsig à la manière openssl. Noter que les standards imposent sha256

openssl smime -sign -binary -in DB.forsig -out DB.signed -signer KEK.crt -inkey KEK.key -outform DER -md sha256

Qui produit une signature PKCS7 détachée dans DB.signed. Maintenant le placer dans le programme :

sign-efi-sig-list -a -i DB.signed -t 'Jul 21 09 :39 :37 BST 2012' db DB.auth

Pour supprimer une clé, simplement signer un fichier de liste de signature, donc pour produire une mise à jours de variable qui va supprime le PK :

> null.esl

Puis le signer :

sign-efi-sig-list -c PK.crt -k PK.key PK null.esl PK.auth

Utiliser UpdateVars.efi pour l'appliquer :

UpdateVars [-a] db DB.auth

où le flag -a doit être présent si le fichier DB.auth a été créé comme ajout, et absent s'il remplace la variable.