

## RE: CryptGetUserKey fails with 8009000d

**Source:** <http://www.derkeiler.com/Newsgroups/microsoft.public.platformsdk.security/2005-09/0143.html>

---

**From:** lelteto ([lelteto\\_at\\_discussions.microsoft.com](mailto:lelteto_at_discussions.microsoft.com))

**Date:** 09/18/05

Date: Sat, 17 Sep 2005 16:19:02 -0700

GetUserKey returns key handle to either the AT\_SIGNATURE or AT\_KEYEXCHANGE keyPAIR in the given container. When you import a PUBLIC key it is not associated with either – it is just a 'temporary' key. (If you generate or import a secret key you would also get just temporary keys – and they cannot be retrieved with CryptGetUserKey.) Within the SAME session you would just have to use the key handle returned by the CryptImport function. Since temporary keys are NOT saved (once you closed the context handle with CryptReleaseContext) next time you need the same key you have to re-import it again. In fact, if you just need to import the public key than you can simple use a temp crypto context by calling CryptAcquireContext with CRYPT\_VERIFYCONTEXT.

Laszlo Elteto  
SafeNet, Inc.

"Daniel" wrote:

> *Hi All.*  
>  
> *I have written a Active X DLL in VB to take care of some cryptography*  
> *functions I need performed by my ASP app (actually it modifies a nice little*  
> *class I found somewhere.)*  
>  
> *The only functions I need the DLL to perform are impoting and signing*  
> *strings with a private key and importing a public key and encrypting*  
> *strings.*  
>  
> *The first problem I had was getting CryptAcquireContext to work on the*  
> *TARGER SERVER because the security context that the application was running*  
> *in did not have access to a use profile. Thanks to Microsoft Q238187 I was*  
> *able to get a container like so:*  
>  
> '  
> ' *try to acquire the existing container*  
> '  
> *lngReturnValue = CryptAcquireContext(hCryptProv, "Container" & vbNullChar,*  
> *SERVICE\_PROVIDER, PROV\_RSA\_FULL, \_*  
> *CRYPT\_MACHINE\_KEYSET)*

> *If lngReturnValue = 0 Then*  
> '  
> *' Try to make a new key container*  
> '  
> *lngReturnValue = CryptAcquireContext(hCryptProv, "Container" &*  
> *vbNullChar, SERVICE\_PROVIDER, PROV\_RSA\_FULL, \_*  
> *CRYPT\_NEWKEYSET Or*  
> *CRYPT\_MACHINE\_KEYSET)*  
> *If lngReturnValue = 0 Then Err.Raise Err.LastDllError, , "DLL error*  
> *code shown above. Error during CryptAcquireContext for a new key container."*  
> *End If*  
>  
> *The first call succeeds. I am never able to successfully make a new*  
> *container.*  
>  
> *However after that I am able to successfully call decrypt and import the*  
> *signature key (using CryptImportKey) and use it to sign some text ( using*  
> *CryptCreateHash and CryptSignHash)*  
> *there is no problem.*  
>  
> *The problem arises when I attempt to import a Public Encryption key.*  
> *CryptImportKey succeeds fine, but CryptGetUserKey fails with error 8009000d.*  
>  
> *I think the problem obviously has something to do with Calling*  
> *CryptAcquireContext with CRYPT\_MACHINE\_KEYSET.*  
>  
> *If I edit the account used for anonymous access on the page and set it to an*  
> *account that is logged in (like mine) then I don't need to use*  
> *CRYPT\_MACHINE\_KEYSET I can create a new*  
> *container with CRYPT\_NEWKEYSET and everything works fine.*  
>  
> *Any help would be greatly appreciated.*  
>  
>  
> --  
> *Daniel*  
>  
>  
>