

Re: Parsing binary values in the registry

> of your byte array, then step through the byte array,
> getting every second byte to skip the 00 bytes. Each
> of those is converted to an ANSI character and added
> to the A2 array. Once that's done you need only use
> Join on the A2 array to get your string.
>
> I'm not sure if that answers the first question. It sounds
> like you want to split an array of characters, but you
> say "an array of 32 length variables". What does that
> mean?
> ---
> ---
> Jim Gregg <jpgregg@xxxxxxx> wrote in message
> news:#uD72erfFHA.3940@xxxxxxxxxxxxxxxxxxxxxxxxxxxx
>> Hello all,
>>
>> I have 2 questions if anyone can assist. First, I am wondering what is
>> the
>> best way to split a variable length value into an array of 32 length
>> variables. The size of the initial array can change from 32 to whatever.
>> Also, there is no splitting character such as tab or space. The initial
>> variable is just a series of values.
>>
>> Second, how can I take a variable length value and split it into an array
>> containing variables that have 4 characters. So, I need to take that long
>> value, take the first 4 characters, put that in a variable, go back to
>> the
>> original value, get the next 4, rinse and repeat. These 4 charatcer
>> values
>> are hex to unicode conversion, so I need to take the first two characters
>> from each 4 digit block and convert it to decimal and finally a
>> character.
>> Basically, I am trying to build a file path from a binary value in the
>> registry. Here is some example code:
>>
>> It is a bit of a mess at this time as I am just trying to write sections
> and
>> then bring it all together at the end:
>>
>> Const HKEY_CURRENT_USER = &H80000001
>>
>> strComputer = "."
>> Set StdOut = WScript.StdOut
>>
>> Set oReg=GetObject("winmgmts:{impersonationLevel=impersonate}!\\\" & _
>> strComputer & "\root\default:StdRegProv")
>>
>> *****
>> This section gets the default profile in Outlook
>> *****
>>

Re: Parsing binary values in the registry

```
>> strKeyPath = "SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows
> Messaging
>> Subsystem\Profiles"
>> strValueName = "DefaultProfile"
>>
>> oReg.GetStringValue HKEY_CURRENT_USER,strKeyPath, _
>> strValueName,strValue
>>
>> strProfile = strValue
>> StdOut.WriteLine strProfile
>>
>> '*****
>> "This section gets each of the keys that hold PST file paths
>> '*****
>>
>> strProfilePath = "SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows
>> Messaging Subsystem\Profiles\" & strProfile &
>> "\9207f3e0a3b11019908b08002b2a56c2"
>> strProfileName = "01023d00"
>>
>> oReg.GetBinaryValue HKEY_CURRENT_USER,strProfilePath, _
>> strProfileName,strValue2
>> StdOut.WriteLine strProfilePath & "\" & strProfileName
>>
>>
>> For i = lBound(strValue2) to uBound(strValue2)
>>
>> If Len((Hex(strValue2(i)))) <> 2 then
>> strPathKeys = strPathKeys & "0" & Hex(strValue2(i))
>>
>> Else
>> strPathKeys = strPathKeys & Hex(strValue2(i))
>> End If
>> Next
>>
>> stdOut.WriteLine Len(strPathKeys)
>> stdOut.WriteLine strPathKeys
>>
>> '*****
>> "This section breaks the above info into 32 bit values
>> '*****
>> No Code yet
>> '*****
>> '*****
>> "The below section reads the reg keys that contain the binary version of
> the
>> file path
>>
>> strPSTPath = "SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows
> Messaging
>> Subsystem\Profiles\" & strProfile & "\" & Left(strPathKeys, 32)
```

Re: Parsing binary values in the registry

```
>> strPSTKey = "001f6700"
>>
>> oReg.GetBinaryValue HKEY_CURRENT_USER,strPSTPath, _
>> strPSTKey,strValue3
>> StdOut.WriteLine strPSTPath & "\" & strPSTKey
>>
>> For i = lBound(strValue3) to uBound(strValue3)
>>
>> If Len((Hex(strValue3(i)))) <> 2 then
>> strPSTKeys = strPSTKeys & "0" & Hex(strValue3(i))
>>
>> Else
>> strPSTKeys = strPSTKeys & Hex(strValue3(i))
>> End If
>> Next
>>
>> stdOut.WriteLine Len(strPSTKeys)
>> stdOut.WriteLine strPSTKeys
>> '*****
>> 'Testing Hex to Decimal, then Decimal to CHR
>> decimal = CLng("&H" & left(strPSTKeys,2))
>> stdOut.WriteLine decimal
>> stdOut.WriteLine Chr(decimal)
>>
>>
>
>
```

• **References:**

- ◆ **[Parsing binary values in the registry](#)**
 - ◇ From: Jim Gregg
- ◆ **[Re: Parsing binary values in the registry](#)**
 - ◇ From: mayayana
- Prev by Date: **[Re: Dynamic variable](#)**
- Next by Date: **[Recurse issue – 2nd try](#)**
- Previous by thread: **[Re: Parsing binary values in the registry](#)**
- Next by thread: **[Re: Parsing binary values in the registry](#)**
- Index(es):
 - ◆ **[Date](#)**
 - ◆ **[Thread](#)**