

How_to_convert_IPv4_address_to_integer_array

This example shows how to convert an IP address in dotted string format to an integer array.

PRECONDITIONS

A buffer containing the IP address in dotted format

Add following lines in .cpp file.

```
TBuf<100> yourIPContainingBuffer;
yourIPContainingBuffer.Append(_L("123.456.789.000"));
TInt length = yourIPContainingBuffer.Length();
RArray<TInt> yourDottedArray;
TInt startPos = 0;
TInt endPos = 0;

for(TInt index = 0; index<length+1; index++)
{
    if(endPos!=length && TChar(yourIPContainingBuffer[index])!=TChar('.'))
    {
        endPos++;
    }
    else
    {
        // You should copy everything starting from startPos to endPos into yourDottedArray
        TInt temp = 0;
        TInt tempLength = endPos-startPos;
        TBuf<4> tempBuf;
        tempBuf = yourIPContainingBuffer.Mid(startPos, tempLength);
        TLex tempLex(tempBuf);
        if(tempLex.Val(temp) == KErrNone)
        {
            // We got our first val, lets put it on the yourDottedArray
            yourDottedArray.Append(temp);
        }
        endPos++; //To accomodate the dot we got
        startPos = endPos;
    }
}
```

While using the array, after each index, keep appending the '.' in the calling code to convert it back into a dotted IP address format buffer/string.

POSTCONDITIONS

An integer array containing the IP address

Added by - Mayank on 19/05/2009

A Simpler and Faster Solution

How_to_convert_IPv4_address_to_integer_array

A much simpler and faster solution to the problem mentioned above exists using the TInetAddr class. Below are two functions that convert an Ipv4 ip address in string representation to integer and vice versa:

```
TUint32 InetAddrV4_StringToIntegerL(const TDesC& aIpDes)
{
    TInetAddr addr;
    User::LeaveIfError(addr.Input(aIpDes)); //leave if ip address is malformed
    return addr.Address(); //returns the integer representation
}

void InetAddrV4_IntegerToString(TUint32 aIpAddr,TDes& aIpDes)
{
    TInetAddr addr;
    addr.SetAddress(aIpAddr); //set the ip address of TInetAddr
    addr.Output(aIpDes); //convert the ip address to string representation
}
```

Following code can be used to access the individual components of the Ipv4 address

```
TUint32 address=InetAddrV4_StringToIntegerL(_L("192.168.0.1"));
TUint32 (byte)address&0xff000000)>>24; //returns 192
TUint32 (byte)address&0xff0000)>>16; //returns 168
TUint32 (byte)address&0xff00)>>8; //returns 0
TUint32 (byte)address&0xff); //returns 1
```

Added by Vaibhav Jain- 21 May 2009