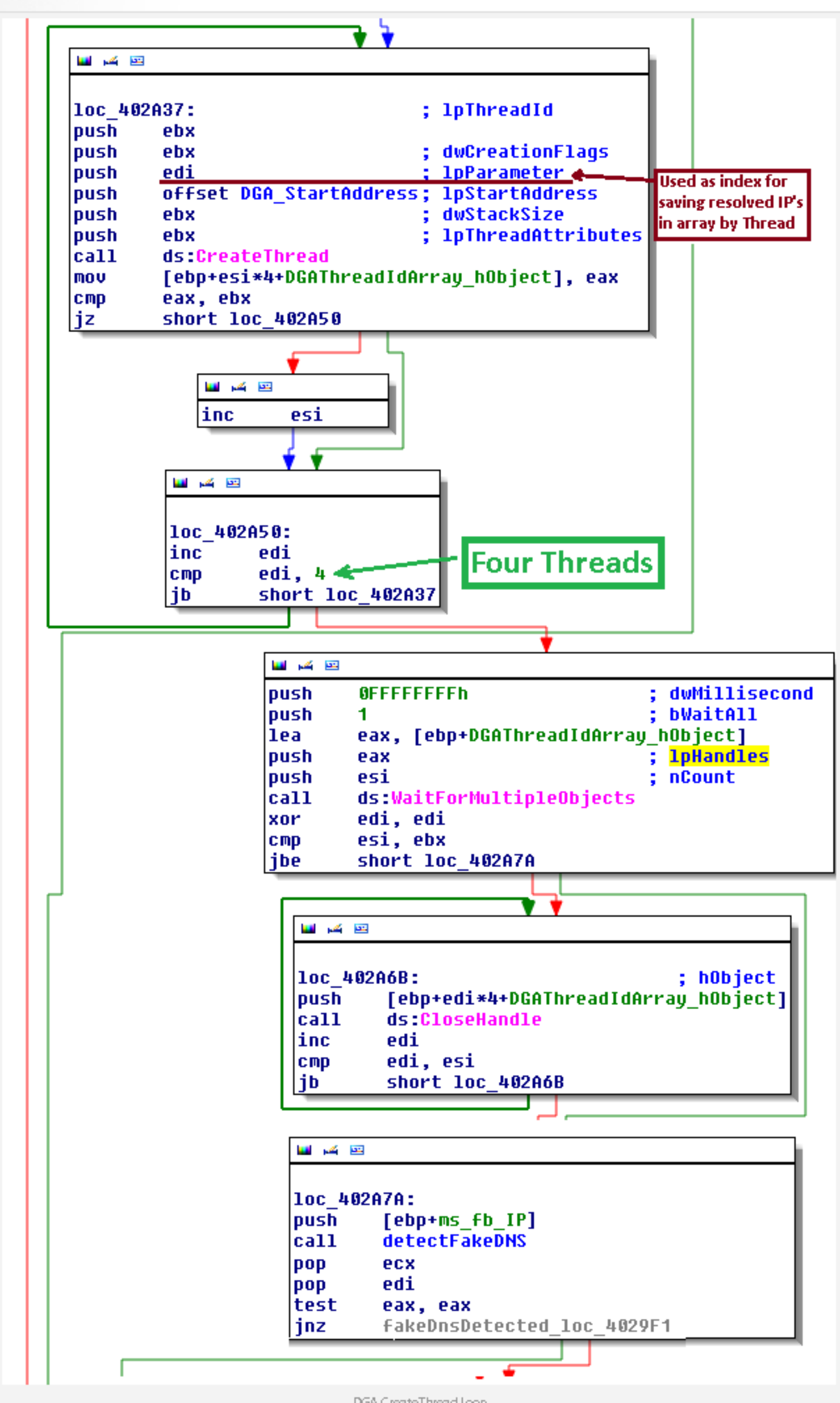
Necurs DGA

In this blog we will discuss **DGA (Domain Generation Algorithm)** of Necurs Malware as i promised in my previous blog. So what are be waiting for lets dig into technical details.

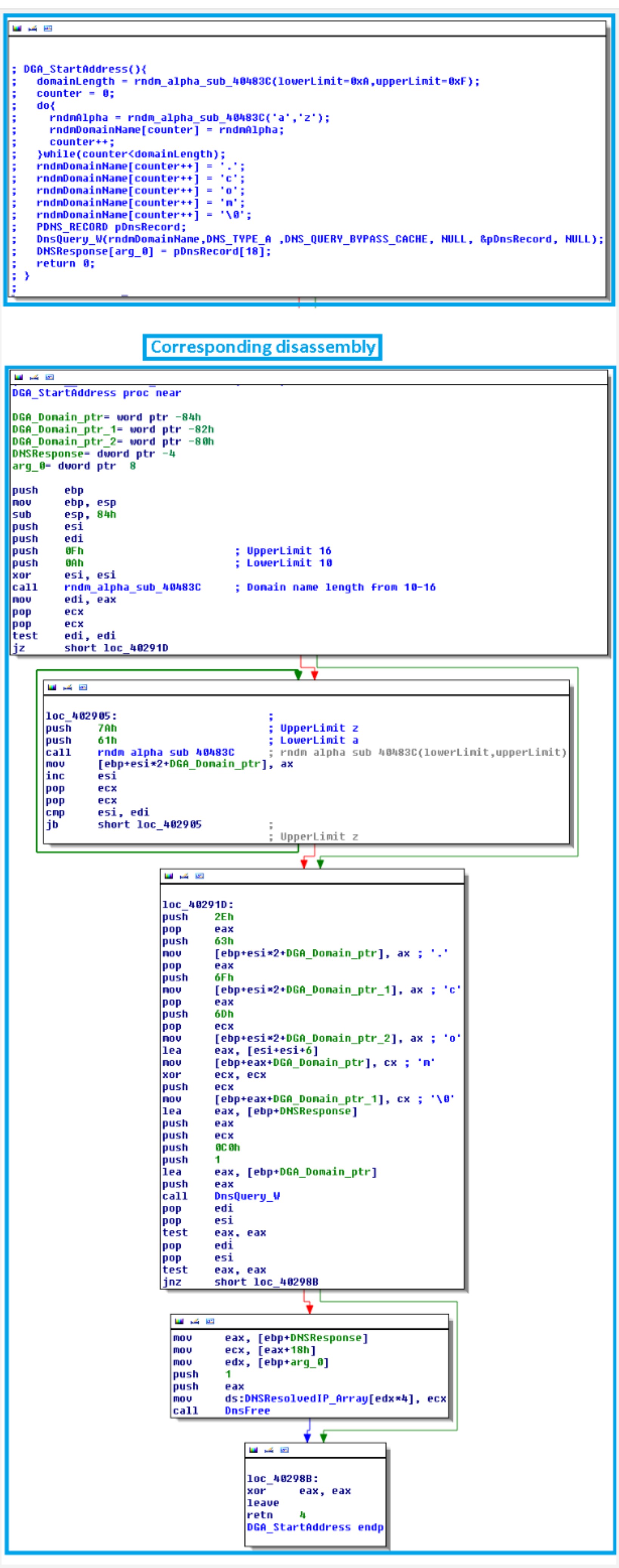
First of all it checks for internet connectivity by randomly connecting with either *'microsoft.com'* or *'facebook.com'* if successful then It saves resolved IP in variable and creates four domains using DGA. It creates four threads of DGA in loop and then waits for those to return:



DGA CreateThread Loop

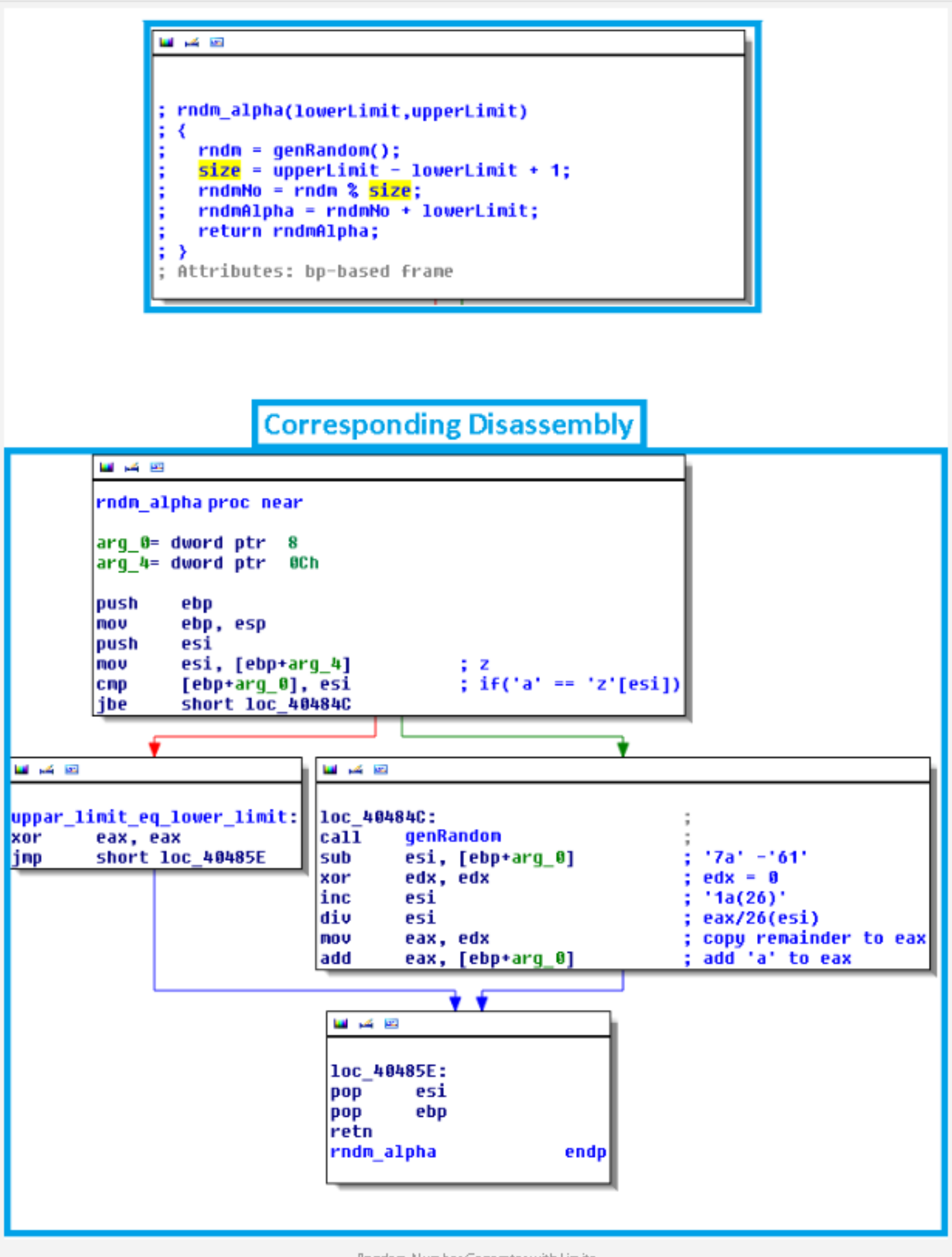
Then Each instance of DGA will return one random domain name in lower case letter of length between 10 to 16 letters and '.com' as TLD (Top Level Domain).

DGA function will call a custom random number generator function with upper and lower limit of 16 and 10 and returned value will be used as length of domain. Then loops through random number generator function calls for earlier returned length number of times with 'a' and 'z' as lower and upper limits. Following is the dis-assembly of the code used to generate domains. After generating it will append '.com' at end of generated string and try to Resolve that domain using Windows API 'DnsQuery_W'. If domain successfully resolves it saves IP address in an Array using a variable passed to Thread as index. DGA code is shown in figure:



DGA Necurs

Random number generator function does not do anything special it just calls another random generator function that generates 32 bit random number without any limit and takes that number adjusts it within limit by diving that by "UpperLimit – LowerLimit' and adding LowerLimit to that and returns that value. Code for this random number generator is shown below:

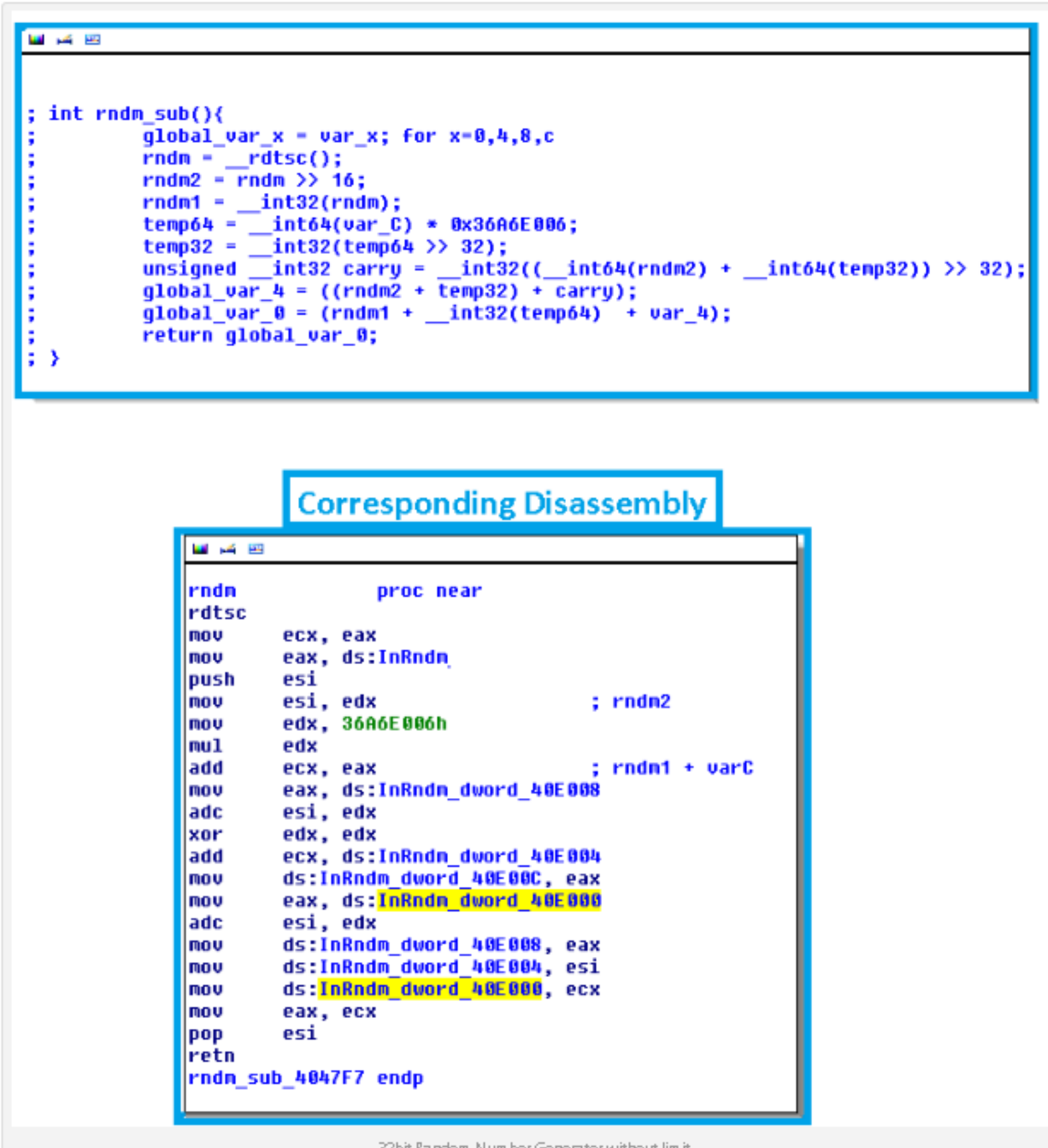


Random Number Generator with Limits

Last random generator without limits uses "**RDTSC**" instruction and four Global variables to generate random number. Values of these variables change with every call to This function. Initial values of those variables are:

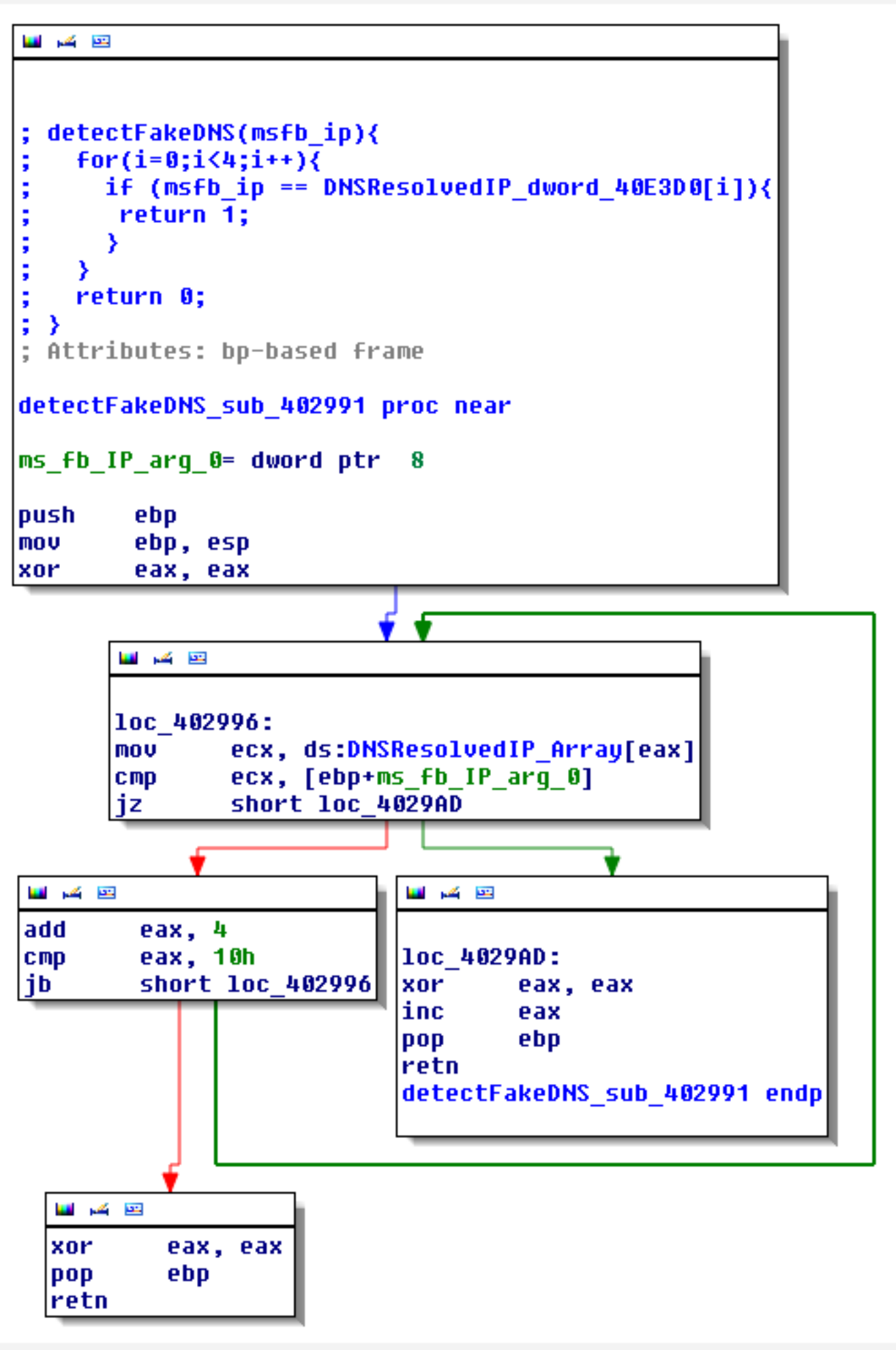
Initial value of variables

Here is code for random generator:



32bit Random Number Generator without limit

After resolving generated domain it simply does not start communicating with those. As expected it has more **trick to avoid analysis**. It compares each IP in resolved IP array with earlier resolved IP of *"microsoft.com"* or *"facebook.com"*. if any IP matches then this function just exits returning zero. This function probably detects '**FakeDNS**' tool.



FakeDNS detect

This is for now in subsequent posts we will analyze communication with Command server.

This entry was posted by Atinderpal Singh on October 18, 2013 at 10:33 am, and is filed under Uncategorized. Follow any responses to this post through RSS 2.0. You can leave a response or trackback from your own site.

Leave a Reply

Logged in as Atinderpal Singh. Log out?

Comment

You may use these HTML tags and attributes: `<a href="" title=""> <abbr title=""> <acronym title=""> <b> <blockquote cite=""> <cite> <code> <del datetime=""> <em> <i> <q cite=""> <strike> <strong>`

Post Comment