

Forum:[Hack Forums](#)

Thread: [\[Codecs\]\[C#+VB\] Stream Library, Speedup your RemoteDesktop for RAT's and more!](#)

DragonHunter - June 17, 2014, 12:37 pm

Hey guys what's up,

I thought of sharing this as I'm quitting the RAT development and focussing at more useful projects. So what is this StreamLibrary? Well it's a library which contains codecs (or so I call them atleast Codecs) that will convert any image/video into a smaller format. It will compress your video (or Remote Desktop if you will) to a smaller size so it will speed up your remote desktop by ALOT

PS: I'm just taking RAT development as a example because this is HF after all.

This can be used in either C# or VisualBasic .NET. What most of the people do in RAT development really is sending the image over (compressed hopefully) and that's it, Or they simply use some motion detection (which is used in the more advanced RAT's) in the hope that it will speedup, which it does.

The codecs I wrote are made for speed and support, you'll find a list below with the features and codecs you're able to use.

The codecs are made for EASY-TO-USE, which means with a stable network you're able to get amazing speed and performance.

Most codecs I wrote are able to handle easily 5760x1080 resolution (which I tested it at) with ~70FPS in speed.

The performance depends at the hardware you're running it at, note: this does not have hardware acceleration

Well ok, after all that talk let's go to the goodies, Ok so this is just an example of an video running at ~30FPS, encoding it at ~80FPS and decoding it at ~100FPS. This is at a full resolution of 1920x800 !!

Features:

- [Driver supported] yes my Unsafe Codecs do support drivers!
- [Unsafe Codec support] Use Pointers to gain more performance!
- [Managed Codec support] use the managed Bitmap/Image classes
- [Multiple Codecs to chose from] There are 5 Unsafe and 6 Managed Codecs
- ... will write down soon more

So where can you get this? Take a look at my GIT <https://github.com/AnguisCaptor/StreamLibrary>

Ok so.. you're having the codec library compiled, so now what ?

So first of all, what is the actual codec you want to use ? If you want it plain and simple you could go for MJPG Codec, which is easy to use and manage but

not really a good thing for Remote Desktop and compression but easy to use non the less

So what if I'm having a RAT and I want the best stability and performance ? use the "UnsafeStreamCodec" which I used personally and is really fast
Having a driver ? no problem the "UnsafeStreamCodec" even does support driver-level!

The Different Managed Codecs

DirectDriver Codec:

I barely used this as it was a long time ago, I would have to look at the code/debug again but I guess it does what the name says

MJPEG Codec:

This is a very simple MJPG Codec which can even be used for real life situations, webrowsers etc, It is using the exact same method

QuickCachedStream Codec:

This will cache all the image(s) and trying to quickly process it to keep the performance up, this codec will use cache in the hope to drop ALOT of data to gain higher performance over the internet
Keep in mind that this codec could be unstable or not working correctly, needs more testing

QuickStream Codec:

The name says what it will do, it will quickly stream your images (remote desktop, animations) so fast as possible with plain simple Motion Detection

SmallCachedStream Codec:

This codec is a modified version of the "QuickStreamCodec" but with caching in the hope it will gain performance over the internet

SmallStream Codec:

I don't know exactly what the SmallStream did, I have to look at it again

The Different Unsafe Codecs

UnsafeCache Codec:

I would need to look up what it does again but I'm pretty sure the name explains it all

UnsafeCachedStream Codec:

I would need to look up what it does again but I'm pretty sure the name explains it all

UnsafeOptimized Codec:

It could be stable or not need to test this completely,
But the idea behind this codec is to really optimize the performance for if there are any video's running for example in Remote Desktop
Or specific animations running in the video, it is able to detect multiple video's running in image(s) and to optimize and focus performance at those animation/video's in the image
It sounds complex perhaps, but it will speedup video a little bit if there is any running

UnsafeQuick Stream:

This is an unsafe codec version as the "QuickStreamCodec", I think

UnsafeStream Codec:

This is a very good choice and even RECOMMENDED to use for everything as this is the most

stable codec and the most reliable codec I made which supports Drivers and more
It does not contain any caching but yet again, It is really recommended to use over any of the other codecs that are in the library

So how do I even use these codecs ?

The codecs are being made for EASY-TO-USE remember ? well the unsafe codecs are all using the exact same methods and same goes for the Managed codecs

An example of using the Unsafe codec (UnsafeStream Codec):

You really need to understand programming to understand calling a bunch of methods

Ok so this is a example to encode a image to bytes using a codec

PHP Code

```
<?php
IUnsafeCodec UnsafeCodec = new UnsafeStreamCodec(80); //initialize our Codec
Bitmap bmp = (Bitmap)Bitmap.FromFile("Some file");
// It's just an image, could be of RemoteDesktop, video, anything, it's a image
int width = bmp.Width;
int height = bmp.Height;
BitmapData bmpData = bmp.LockBits(new Rectangle(0, 0, bmp.Width, bmp.Height),
ImageLockMode.ReadWrite, bmp.PixelFormat);
//since it's a unsafe codec we need the LockBits to grab the pointer

using (MemoryStream stream = new MemoryStream())
{
    UnsafeCodec.CodeImage(bmpData.Scan0,
        new Rectangle(0, 0, width, height), //the area of the image to use
        new Size(width, height), //give the total size of the image
        bmp.PixelFormat, //the format of the image
        stream); //where the bytes will be stored in
}
```

Ok so the bytes are stored now in "stream" right, now what can be do with the data(bytes) in stream ?

The thing that pops to my mind atleast is sending it over the internet (like remote desktop does in RAT's right...)

And then let's decode the bytes to a image so we're able to see something again ?

To decode bytes to a image is actually really simple, well how simple ? only 2 lines of code!

PHP Code

```
<?php
IUnsafeCodec UnsafeCodec = new UnsafeStreamCodec(80); //initialize our Codec
Bitmap DecodedImage = UnsafeCodec.DecodeData(new MemoryStream(
YOUR_BYTES_OR_STREAM)); //where is "YOUR_BYTES_OR_STREAM" should be what it says
```

And that is how you decode from bytes which was provided by the Encoder, back to an Image again using the Decoder

More technical stuff to keep back in your head:

Ok so when using the Managed/Unsafe codecs you should remember that when using these you need a Encoder and Decoder variable (both initialized ofcourse)

And decoding and encoding in the order you encoded it in

So when doing Encode, you should decode

Wrong example: Don't decode twice the same data! like so

```
Encoded = Encode(...);
```

```
Decoded = Decode(Encoded);
```

```
Decoded = Decode(Encoded);
```

Decoding something twice in a row maybe could cause problem, I did not test this completely but It might will!

I hope I've wrote it detailed enough for people to understand otherwise I would be happy to answer questions,

And please don't bash this thread like "OH THIS COMPLETELY RUINED MY RAT SALE" or like "THE CODE IS HAVING SO MANY BUGS LOL"

Some codecs are simply not 100% stable and when you read above about the codecs you know which 1 to use >>UnsafeStream Codec<<

Some of the codecs are simply not 100% finished as it didn't worked out with performance or some other issue

Just enjoy the library and the features it's giving you

Find any bugs ? share them at GIT or simply PM me

Programs/Projects using my StreamLibrary:

Have a good one ?

Sho?kwave - June 17, 2014, 12:40 pm

The amount of work put in this is phenomenal, I still can't believe you would release something this huge, I hope people give credit when they use this.

tweety - June 17, 2014, 12:48 pm

You should have sold this instead of releasing it to a community that would rather abuse it, but won't learn from it. Inb4 20 new RAT's in the market section using this as a base.

Nextrum - June 17, 2014, 12:50 pm

I agree with Tweety, this is probably worth quite a bit to the right person. However, this looks phenomenal and I'll look into possibly implementing it into my Movie Streamer app. Keep up the amazing work!

TheBarret - June 17, 2014, 12:59 pm

(Jun 17, 2014 12:48 PM)Tweety. Wrote: [\[CITE\]](#) You should have sold this instead of releasing it to a community that would rather abuse it, but won't learn from it. Inb4 20 new RAT's in the market section using this as a base.

I kinda agree with Tweetz on this, you could rly make some good deals with this work. However making it public is also very nice.

Pro's mate, good job.

Meerkat.? - June 17, 2014, 1:01 pm

Wow. This is beyond fucking awesome.

Temploit - June 17, 2014, 1:04 pm

(Jun 17, 2014 01:01 PM)Mavamaarten~ Wrote: [\[CITE\]](#) Wow. This is beyond fucking awesome.

Couldn't have said it any other way. Amazing job!

TheBarret - June 17, 2014, 1:06 pm

(Jun 17, 2014 01:04 PM)Temploit Wrote: [\[CITE\]](#) Couldn't have said it any other way. Amazing job!
So it seems the Dutch play good match of football next to having good Dj's, adding "good coders" to the list XD

SIM-1676753 - June 17, 2014, 1:12 pm

This is beyond most of the skiddy shit posted here. Awesome release as always!

He B TeMy - June 17, 2014, 1:36 pm

This is so good, gonna test it's speed soon !

Thank you !!

DragonHunter - June 17, 2014, 1:54 pm

Thanks for the awesome posts ??

I updated the sample project to capture your screen and encode/decode to show how to do it ?
At my own pc the GDI Screen Capture is only at ~10FPS, not really the best screen capture for my pc but it never worked great for me ?

Atleast the encode/decode is really really fast
Also checkout the resolution I'm testing it with, 2560x1440 ?

Only3 - June 17, 2014, 1:56 pm

this lib fit hf as its for ppl that cant do / code it by them self look good but i dont like lib's cls's warp's
Good Work on This

SIM-1456245 - June 17, 2014, 3:24 pm

Wow this is amazing. But it's not worth it for free. You'll see tomorrow 10 RATs with amazing speeds. You should've sold this.

VanToMVB - June 17, 2014, 11:40 pm

That's Awesome man !!!!!
can u make a vb.net copy :v ? ????

Fgownage - June 18, 2014, 12:08 am

You know what would be awesome? Heavens gate.

Vysse - June 18, 2014, 10:11 am

I just finished up my class to do bitmap comparison and now it's useless lol. Seriously though looks amazing.

VirinomTR - June 18, 2014, 11:42 am

Oh my god amazing !! Thanks for sharing it free

DragonHunter - June 19, 2014, 1:40 pm

Made a server / client example, take a look in GIT where the source is ;)

<https://github.com/AnguisCaptor/StreamLibrary/commit/dbed2dc6b0b47800db10a6fc37a48da6205816e6>

Also, I ran this over WAN (yes the big internetz) with a resolution of 2560x1400
FPS was around average ~14FPS

VirinomTR - June 19, 2014, 3:51 pm

(Jun 19, 2014 01:40 PM)DragonHunter Wrote: [\[CITE\]](#) Made a server / client example, take a look in GIT where the source is ;)

<https://github.com/AnguisCaptor/StreamLibrary/commit/dbed2dc6b0b47800db10a6fc37a48da6205816e6>

Also, I ran this over WAN (yes the big internetz) with a resolution of 2560x1400
FPS was around average ~14FPS

Oh my god you are a real hero :) Thanks again really nice project

VanToMVB - June 19, 2014, 11:23 pm

why it does not capture the Taskbar & mouse ????? :(

VirinomTR - June 20, 2014, 6:56 am

(Jun 19, 2014 11:23 PM)VanToMVB Wrote: [\[CITE\]](#) why it does not capture the Taskbar & mouse ????? :(

View [this](#) article. U have to draw cursor on to screenshot. There is no way to capture it directly. (As i know)

markzuckerberg - June 20, 2014, 8:46 am

Wow Mate, this is awesome! ? i did a rat from about a year ago but with ftp uploads ? .. the server uploads the picture and then the client downloads it, but it was very slow ? .. because it depends on the internet + the hosting bandwidth! This thread that you made helped me with a new projects to do! Well keep going man you're the best ?

Server97 - June 20, 2014, 11:09 am

This looks amazing! User's should pay for this :P Wow! Exceptional work OP!

VirinomTR - June 20, 2014, 11:52 am

(Jun 20, 2014 08:46 AM)GlobalCoder Wrote: [\[CITE\]](#) Wow Mate, this is awesome! ? i did a rat from about a year ago but with ftp uploads ? .. the server uploads the picture and then the client downloads it, but it was very slow ? .. because it depends on the internet + the hosting bandwidth! This thread that you made helped me with a new projects to do! Well keep going man you're the best ?

OMG probably worst way for remote desktop. Why don't use binary formatter de/serialize its quite simple and really fast

SIM-62706 - June 21, 2014, 5:48 pm

man thanks a lot for the source and for the client/server example i think that it's very interesting and i am going to try to check how it works
keep going! :D

The iNfaM0us - June 22, 2014, 8:57 am

This looks cool, I will play around with it when i get home.

VirinomTR - June 22, 2014, 1:18 pm

(Jun 17, 2014 01:54 PM)DragonHunter Wrote: [\[CITE\]](#) Thanks for the awesome posts ??

I updated the sample project to capture your screen and encode/decode to show how to do it ?
At my own pc the GDI Screen Capture is only at ~10FPS, not really the best screen capture for my pc but it never worked great for me ?

Atleast the encode/decode is really really fast
Also checkout the resolution I'm testing it with, 2560x1440 ?

I want to use this in my vb.net application but what is the equivalent of pointer (of byte) in vb.net?
As i know vb.net not allows the use unsafe codes directly. Is there any way to adapt this project to vb.net ?

DragonHunter - June 22, 2014, 1:22 pm

(Jun 22, 2014 01:18 PM)VirinomTR Wrote: [\[CITE\]](#) I want to use this in my vb.net application but what is the equivalent of pointer (of byte) in vb.net? As i know vb.net not allows the use unsafe codes directly. Is there any way to adapt this project to vb.net ?You can just use the IntPtr which is also in visualbasic

VirinomTR - June 22, 2014, 2:46 pm

(Jun 22, 2014 01:22 PM)DragonHunter Wrote: [\[CITE\]](#) You can just use the IntPtr which is also in visualbasic

Thanks also i have problems with fixed statement.Is there any equivalent to vb.net ?

Code fixed (byte* ptr = EncodeBuffer)
Edit:And also this Pointer(Of System.Void)

Thanks so much

ubbelol - June 22, 2014, 2:48 pm

(Jun 22, 2014 02:46 PM)VirinomTR Wrote: [\[CITE\]](#) Thanks also i have problems with fixed statement.Is there any equivalent to vb.net ?

Code fixed (byte* ptr = EncodeBuffer)
Unsafe code isn't supported in .NET. You'll need to use the Marshal class.

VirinomTR - June 23, 2014, 5:43 am

(Jun 22, 2014 01:22 PM)DragonHunter Wrote: [\[CITE\]](#) You can just use the IntPtr which is also in visualbasic

I trying to convert to vb.net but it is more difficult than i thought.
Is it possible to convert it to vb.net ?

ubbelol - June 23, 2014, 5:46 am

(Jun 23, 2014 05:43 AM)VirinomTR Wrote: [\[CITE\]](#) I trying to convert to vb.net but it is more difficult than i thought.

Is it possible to convert it to vb.net ?
Yes it's possible. You just have to know what you're doing. It's gonna be a lot of work though.

DragonHunter - September 6, 2014, 3:10 am

Just a b*u*m*p because I see everyone in this thread they're enjoying it so far, Thanks for the awesome comments ^.^

TheExclusive - November 18, 2014, 8:02 am

Looks very nice gonne be handy soon ! :)

Dark N3BULA - November 18, 2014, 8:24 am

Pure class ! Awesome post ... ty for sharing

idb. - May 17, 2015, 12:33 am

You should write a H.263 codec DH. Will love you forever :D <3

xxdarklord65xx - January 31, 2017, 4:55 am

i really appreciate your work mate .

i had never seen such post or thread like this !

im so happy to use it in my RAT .

but i have some Questions ,

1. I made a test and encode the image and i saved the bytes of the image in a folder and it was 32 KB . the real image was 90 KB

so some of the rats there image size is like 4 to 10 kb . and so fast . is there such thing i missed or did not use it ?

2. I encode and save the image in a stream and Send it Through a tcp channel . and when the server receive the image from the client it decode the image and finally the image is viewed . but its really not that much fast my remote desktop form is kind of laggy and delayed . can u please tell me any ideas or information to improve the Remote Desktop .. cuz i have been trying 3 month to make a perfect RDP

#Kat - February 1, 2017, 10:25 pm

Okay I've ben staring at this thread for a day now OMG this is a huge share man thanks a lot you don't realie how significant it really is .. I think you do tho haha

Cheers!

(May 17, 2015 12:33 AM)idb. Wrote: [\[CITE\]](#) You should write a H.263 codec DH. Will love you forever :D <3

?? it is so close right I was thinking the same exact thing

Sorzus - February 2, 2017, 12:44 am

Hey,

I'm using the concept of the unsafestreamcodec in orcus and I really want to thank you for this project. I use a modified version for webcam and remote desktop. Some improvements I want to note:

Sometimes buffers can be avoided, memmove is a really powerful feature which could be used a few times instead of buffer + memcpy

The memorystream is slightly ineffective (because first you copy the data into the stream and then copy it to an array), from my point of view it would be better to collect all arrays and merge them at the end with blockbuffer.

The bottleneck is the compression. TurboJPG is around 4-6 ms faster in my benchmarks which is quite a lot.

I increased the size of the scan area slightly which results in faster processing but bigger images. I think that this could be faster.

If you are using wpf, you can directly copy the data into the backbuffer of a writeable bitmap and add dirtyrects for the updated areas which is extremely efficient.

For better integration with the desktop duplication api, methods to update and move multiple rectangles can be added which results in skipping the whole scanning process.

In good days, im able to reach around 50 fps with the Codec and the desktop api by Microsoft over encrypted tcp. Cheers.