H.264 Motion Vectors Based Cyber Defense/Attack Techniques

AUTHORS
1. Prof. Ofer Hadar
2. Mr. Rami Segal
3. Mr. Raz Birman

THE PROBLEM

Attackers find the multimedia world in general and video streaming in particular, an attractive backdoors for their cyber-attacks. Multimedia covert channels provide reasonable bandwidth and long lasting transmission streams, suitable for planting malicious information and therefore used as an exploit alternative. Most attack algorithms that use video focus on planting malicious information inside the video headers. Such techniques are easy to detect and provide a limited delivery bandwidth, therefore increasing attackers' motivation to use video payload exploit. In this article we propose a method to protect against attacks that use the video payload. The payload is the part of the transmitted video data that carries the actual picture content viewed by the movie viewers (users). Therefore, it is more challenging to protect against such attacks - nobody can argue with the actual image content.

In this paper, we will demonstrate attacks that take advantage of video payload and in addition propose a set of defense techniques for improving defense efficiency.

KEY WORDS

Exploit	Watermarking	Cyber	Stego objects
Steganography	Covert channels	Discrete cosine transform (DCT)	Motion vectors estimation

BACKGROUND

Estimated Motion Vectors (EMV) are used in streaming video transmission to describe Macro-Blocks' (MBs) estimated movement between video frames. The attack algorithm will take advantage of lack of sensitivity of movie viewers to small deviations of Macro-Block (object) movements from their original path. Viewers are not likely to notice minor noise around moving MBs. Moreover, since the viewer does not know the accurate real position of MBs in the original video movie, they are not likely to notice minor displacement changes that affect MBs position accuracy.

Video compression protocols, such as E.264 for example, are based on estimating MBs movement (represented by EMVs). In order to minimize the number of bits required to represent transmitted EMVs, the algorithm assumes that neighboring MBs statistically move in similar vector values (size and direction). Therefore, it is possible to transmit only the EMVs delta values compared to neighboring MBs. In order to find the EMV delta value, a Median of all neighboring MBs EMV is calculated and only the delta is sent (Error Vector).

OBJECTIVE

The cyber-attack algorithm is based on identifying MBs with relatively small displacement movement compared to other neighboring MBs, and using their displacement content as place holders for malicious data. Common video streams contain many such MBs, thus providing an effective exploit covert channel with reasonable bandwidth potential (kbps).

METHOD

The cyber-attack is based on the criteria of finding all Error Vectors that are lower than a pre-defined threshold value (e.g. |Error Vector| $\leq \sqrt{2}$).

For example, the Error Vectors that are lower than $\sqrt{2}$ are:

(-1,-1)	(0,-1)	(1,-1)
(-1,0)	(0,0)	(1,0)
(-1,1)	(0,1)	(1,1)

These nine (9) Error Vector positions can be used as a dictionary for conveying malicious data. They can represent eight (8) different values (equivalent to 3 bits – 000 - 111) plus one additional value (the center – 0,0). The following table illustrates the 8 + 1 dictionary options corresponding to the above Error Vectors.

7	0	1
6	X	2
5	4	3

Where X can be used as a special character, such as 'New Line', 'Space', etc.

For example, the encoder (or a proxy) that implements the algorithm will search in the frames for Error Vectors which meet the criteria. Let's assume that the algorithm will find an Error Vector (-1,1). Assuming that the attacker wants to transmit the value '2' (010), this Error Vector will be replaced by (1,0).

In the same example, the decoder will search for Error Vectors that satisfy the same criteria. Once found, it will identify the (1,0) Error Vector and will convert it to the appropriate dictionary value (in this example - '2').

Our research and experiments demonstrate that the Error Vector threshold criteria of $\sqrt{2}$ can support a covert channel transmitted bandwidth of ~3kbs. This bandwidth was calculated according to the following formula:

$$B = \frac{p * Frame\ Rate}{Total\ Number\ of\ frames} \sum_{i=1}^{N} (Total\ moving\ macrobloks\ per\ frame)$$

Where p is the number of bits that represent each element in the dictionary.

The Defense algorithm

Our proposed defense algorithm consists of the same algorithm as above. The idea is to change the Error Vector which meet the criteria randomly.

EXAMPLES

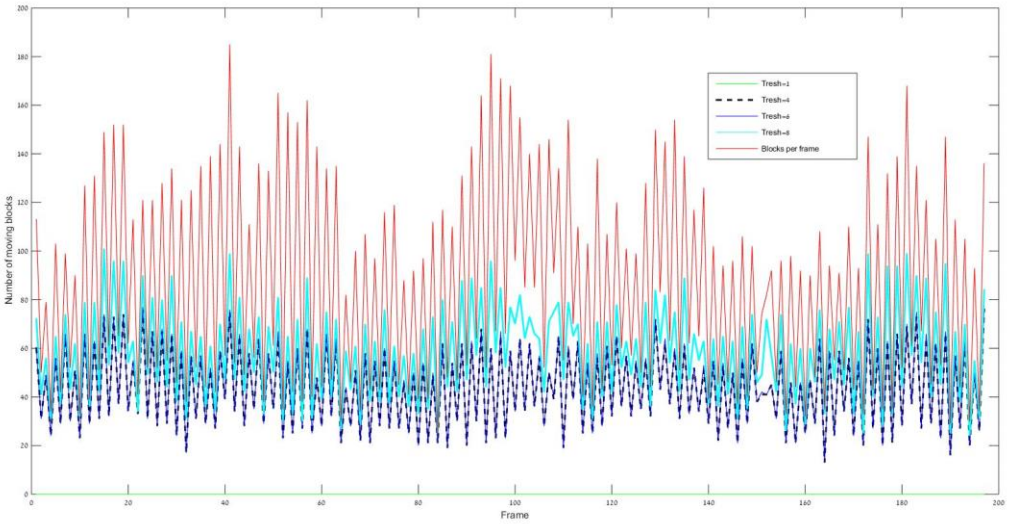
After investigating this algorithm on one specific video file (frames: 100, size: 360 x 640, frame

rate: 25 frame/sec, overall bitrate: 2128 kbps, format: AVC/H.264, color space: YUV, Chrome subsampling: 4:2:0, Bit depth: 8 bits) we found out that we can create a cover channel inside this video with more than 3Kbps bandwidth. Changing threshold (changing the number of selected Error Vectors base on their size criteria) can change the cover channel bitrate. The value that determines our attack's strength is the threshold level T, by decreasing the T value we get better results for minimum damaging the original video (has a slight effect on the video quality) while reducing the cover channel bandwidth. At the same time, this strong implementation has a slight effect on the video quality if we are increasing the threshold to higher values (more Error vectors are selected). In this case the results show that we can achieve higher covered channels bandwidth while keeping reasonable video quality.

The T value can be modified for different purposes. Where security is more important than image quality we would suggest to use a high T value, and where the security has less importance than the video quality we suggest to use a low T value.

In this picture we demonstrate the macro-blocks that has any value of Error Vectors (the Red blocks) and macro blocks that has Error vectors with a size less than $\sqrt{2}$ – Yellow macro blocks (in this case criteria is $|\text{Error Vector}| \leq \sqrt{2}$)

Graph showing the smallest amount of vectors revalued certain threshold in each frame.



The overall defense algorithm: After we inspected the effect of our technique on the video, we were able to create a cost –effectiveness graph of our entire defense algorithm, shown in Figure 21, for creating this graph We can set different parameters for each of the threshold criteria, from delicate Error vectors changes to rough changes. This approach allows the users of the scheme to make risk analysis of their systems and set the acceptable quality degradation based on the cover percentage.

NOVELTY

In this short article we demonstrate how attacker can create cover channel with very reasonable bandwidth by exploiting estimated motion vector in H264 video stream. This technique can be done via proxies in near real time without the need to decode the all video there for it will consume very low CPU processing time and can become very attractive to attacker. The protection technic is very simple and using the same attack technique to insert random noise in the client side. The client can choose his protection level. If he trust the video source he can decrease the T level and to improve his video quality. If it is a suspect source, and the user insist to view this video content he will increase the T value (while reducing the video quality) and will be able to watch it safely.