Analysis of the Nagravision Video Scrambling Method

Markus G. Kuhn*

23 August 1998

Abstract

The Kudelski *Nagravision* pay-TV conditional access system can be practically broken by image processing algorithms that rearrange the lines of a field based on statistical properties of typical TV images. With some knowledge about the limitations of the scrambling hardware one can reconstruct the scrambled TV image in real-time without knowledge of the cryptographic secret stored in the subscriber smartcard.

Draft Technical Report $Id: nagra.tex,v 1.11 2000-10-11 11:40:53+01 mgk25 Exp $

1 Introduction

Pay-TV broadcasters employ conditional access systems to ensure that only TV viewers who have payed a subscription fee and who have in return received a decoder box can watch the TV channel. The *Nagravision* [1] conditional access system for PAL television developed by Kudelski SA, Cheseaux, Switzerland, is used for instance by the pay-TV broadcasters *Premiere* (Germany), *Teleclub* (Switzerland), *Canal+* (France, Spain), and *Cinemania* (Spain). Like with other hybrid video scrambling systems such as *EuroCrypt* [2, 3] or *VideoCrypt* [4], *Nagravision* sends a digitally encrypted control word over the radio interface to the decoder in order to control the descrambling of an analog TV signal. The control word is decrypted in a

*University of Cambridge, Computer Laboratory, Pembroke Street, Cambridge CB2 3QG, United Kingdom. Email: mkuhn@acm.org

smartcard and converted into the seed value for a random number generator. This random number generator then controls the image descrambling process for the next few seconds. *Nagravision* scrambles the video signal by permuting the lines within a field. It also inverts the audio spectrum by mixing it with a 12.8 kHz sine wave carrier to make it unrecognizable. The audio signal can trivially be descrambled by just inverting its spectrum a second time; it is not protected by any cryptographic mechanisms.

Like with all hybrid scrambling systems, which digitally control the scrambling of a video signal that is transmitted in analog form, there are two different classes of techniques for descrambling the video signal without using a regular decoder or smartcard:

- Microelectronics testing equipment can be used to extract the decryption algorithm and secret key data from the smartcard and with this knowledge compatible pirate smartcards and decoders can be manufactured [5].
- Properties of typical TV signals can be used to reconstruct the original image or the random number seed value that controls the descrambler which is then used to descramble the entire image in high quality [6, 7]. This technique makes it unnecessary to break the digital cryptography or smartcard security aspects of the system and it can be implemented without using any genuine decoder hardware.

The first type of attack can also be used on fully digital systems such as DVB, which encrypt an MPEG compressed TV signal. The second type of attack, on which we focus here, is not any more possible with digital conditional access systems.

2 Video Scrambling

The B,G/PAL TV standard [8, 9], which is used for instance in Germany, displays 25 frames per second. Each frame is displayed as a sequence of two interlaced fields so that the screen image is updated with a rate of 50 fields per second. With B,G/PAL 15625 lines are displayed per second, which leaves 64 μs per line. Around 52 μs of this line interval contain active line content; the remaining time is the horizontal blanking interval, which consists of a 1.55 μs front porch, a 4.7 μs sync pulse and a 5.8 μs back porch. Of the $15625/25 = 625$ nominal lines that are transmitted per frame in 1/25 s,

the frame line number intervals 23–310 and 336–623 each contain the $288 = 2^8 + 2^5$ visible lines of one field. In the following, we will only talk about the lines within a single visible field and we will use field line numbers ranging from 0 to 287 to refer to these lines. The remaining $625 - 2 \cdot 288 = 49$ lines of a frame do not contain a visible image signal and form the vertical blanking intervals. These are used for the vertical sync pulse and for digitally transmitted data such as videotext and control signals for conditional access systems.

Nagravision scrambles the image by permuting lines within a field. In addition, the boundaries between fields are shifted by 32 lines between the scrambled and descrambled image as Fig. 1 shows. Line 287 of any field is not affected by *Nagravision*. The last 32 lines (field lines 255–286) of one scrambled field and the first 255 lines (field lines 0–254) of the following field together form a group of 287 lines that are permuted and then used together to form a single field in the descrambled signal. The decoder sends out the first line of the descrambled field while the 33rd line of the scrambled field is being received. This line scrambling can be described by a permutation function $p : \{0, \ldots, 286\} \to \{-32, \ldots, 254\}$ which says that field line i from the clear image appears as field line $p(i)$ in the scrambled signal. A negative field line number $p(i)$ refers to field line number $287 + p(i)$ in the previous field. The descrambling permutation $p^{-1} : \{-32, \ldots, 254\} \to \{0, \ldots, 286\}$ is just the inverse function of p such that line $-32 \leq j \leq 254$ in the scrambled field can be found as line $p^{-1}(j)$ in the descrambled signal. The 4.43 MHz PAL color burst signal in the back porch is not permuted. However, in the SECAM variant of *Nagravision*, the unmodulated 4.406 or 4.250 MHz chroma subcarrier in the back porch is permuted together with the rest of the line.

Nagravision permutes the 287 lines that will form a field in the descrambled signal by buffering 32 lines in RAM and by writing and reading lines into and out of this buffer in a pseudo-random order. We shall refer to these 32 buffer lines as $B_0, \ldots, B_{31}$. While field line number i is being received, the content of buffer $B_{v(i)}$ is sent out as the descrambled signal, and then buffer $B_{v(i)}$ will be immediately overwritten with the signal of the incoming line. The buffer selection function v has the form

$$v(i) = \begin{cases} S(u(i)), & \text{for} \quad 0 \leq i \leq 254 \\ i - 255, & \text{for } 255 \leq i \leq 286 \end{cases} .$$

Using v, we can write the descrambling permutation function as

$$p(i) = \max\{j \mid -32 \leq j < i \wedge v(j \bmod 287) = v(i)\}$$

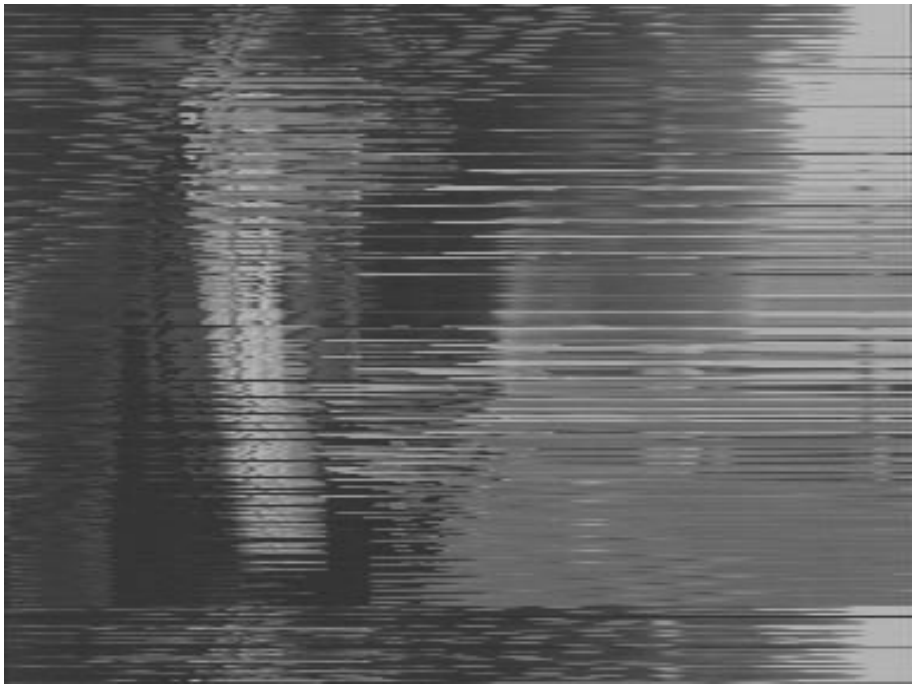

Figure 1: Example of a *Nagravision* scrambled image field.

or equivalently

$$p^{-1}(i) = \min\{j \mid i < j < 287 \wedge v(j) = v(i \bmod 287)\}.$$

$S : \{0, \ldots, 255\} \to \{0, \ldots, 31\}$ is a substitution table stored in non-volatile memory in the descrambler. It is constant over a long time, but it can be updated over the radio interface from time to time.

The function $u : \{0, \ldots, 255\} \to \{0, \ldots, 255\}$ depends on two parameters $r \in \{0, \ldots, 255\}$ and $s \in \{0, \ldots, 127\}$. It has the form

$$u(i) = (r + ti) \bmod 256 \qquad \text{with} \qquad t = 2s + 1.$$

The 15-bit seed value (r, s) changes for every field. It is calculated from the decrypted 64-bit control word that the smartcard sends to the decoder every two seconds.

Since $t = 2s + 1$ has no common factor with 256, the function u is for all combinations of r and t a permutation on $\{0, \ldots, 255\}$. The odd numbers form a multiplicative subgroup in the set of integers modulo 256. This means that for every $i \in \{1, 3, \ldots, 255\}$ there exists exactly one inverse element $i^{-1} \in \{1, 3, \ldots, 255\}$ such that $i \cdot i^{-1} \bmod 256 = 1$. With $2^{15} = 32768$

possible combinations of r and t, there are 32768 different functions u and depending on the structure of S there are up to as many different functions v and p possible with a fixed substitution table S.

To learn more about the relationship between members of the set V_S of all functions v for a fixed S, we first have a look at the structure of the set U of all 2^{15} functions u:

For any pair of functions $u, u' \in U$ with $u(i) = (r + ti) \bmod 256$ and $u'(i) = (r' + t'i) \bmod 256$ there exists exactly one pair of numbers a and b such that $u(a + bi) = u'(i)$. Proof: The numbers $a = (r' - r)t^{-1}$ and $b = t't^{-1}$ do the job, because $u(a + bi) = (r + t(a + bi)) \bmod 256 = (r + t((r' - r)t^{-1} + t't^{-1}i)) \bmod 256 = (r + (r' - r) + t'i) \bmod 256 = (r' + t'i) \bmod 256 = u'(i)$.

Equivalent transformations are possible in the set V_S of functions v. However, there could exist more than one pair of numbers a and b such that $v(a+bi) = v'(i)$ for a given pair of functions $v, v' \in V_S$. This can happen with certain pathological substitution tables S. For instance, if $S(i) = 0$ for all i, then any pair (a, b) will result in $v(a + bi) = v'(i)$ for all i.

The nature of the scrambling method restricts the permutation p by the condition $p(i) < i$ or equivalently $p^{-1}(i) > i$, and by the condition that the sequence $p(0), \ldots, p(286)$ can be split up into 32 monotonically increasing subsequences. Each of these subsequences corresponds to the sequence of lines that were stored in one of the 32 buffers, that is for any $0 \le i < j \le 286$ with $v(i) = v(j)$ we have $p(i) < p(j)$. This leaves $32^{287-32} = 2^{1275}$ possible permutations p if S is not known, compared to only 2^{15} possible permutations if S is known.

3 Attack techniques

The following techniques are based on the observation that in a typical TV image C, the correlation of two pixels drops quickly as the distance between these pixels increases. This means for instance that for two pixel luminosities $C_{x,y}$ and $C_{x',y'}$, the average absolute difference $E(|C_{x,y} - C_{x',y'}|)$ is smaller or alternatively the normalized correlation $E(C_{x,y}C_{x',y'})/\sqrt{E(C_{x,y}^2)E(C_{x',y'}^2)}$ is larger if the two pixels are direct neighbors than if they are many lines apart. The permuted lines can be sorted back into an arrangement close to their original order just by rearranging them in a way that maximizes the similarity (correlation) between neighbor lines.

3.1 Reconstructing the Permutation

Let $C_{x,y}$ be the luminosity or even the whole three-dimensional color vector of the pixel (x,y) in the scrambled field, where as before negative line numbers refer to pixels in the preceding field. Then the matrix $K \in \mathbb{R}^{288\times288}$ shall be the correlation matrix for a field, defined as

$$K_{i,j} = \frac{\sum_k C_{k,i-33} C_{k,j-33}}{\sqrt{\sum_k |C_{k,i-33}|^2 \cdot \sum_k |C_{k,j-33}|^2}}.$$

$K_{i,j}$ is a measure for the similarity of lines $i-33$ and $j-33$. Obviously $K_{i,j} = K_{j,i}$ and $K_{i,i} = 1$, therefore we only have to determine $K_{i,j}$ for all $1 \le i < j \le 288$. Exchanging two lines $i-33$ and $j-33$ in the image C corresponds in K to swapping the contents of the lines i and j, plus swapping the columns i and j. The goal of rearranging the lines of C to form the original image corresponds to permuting lines and columns in K to bring the largest values as close as possible to the diagonal, so that we maximize the value of a profit function such as

$$G(K) = \sum_{i=1}^{287} K_{i,i+1}.$$

This corresponds to finding the permutation matrix P that maximizes the value $G(PKP^\mathsf{T})$. P relates to the permutation p that we want to reconstruct by $P_{p(i)+33,i} = 1$ for all i and all other $P_{i,j}$ are zero.

In an alternative formulation of the same problem, we look at an undirected graph G_K with nodes N_i ($-32 \le i \le 254$), of which each corresponds to a field line i in the scrambled image. The edge connecting N_i and N_j in this graph has the value $K_{i+33,j+33}$ for all i and j. We then look for a Hamiltonian path (i.e., a path that visits all nodes exactly once) of the form $N_{p(0)}, \ldots, N_{p(287)}$, that fulfils the previously stated conditions for p and whose sum of edge labels is maximal. Finding such a path is a variant of the Traveling Salesman Problem [10, 11], which unfortunately is known to be NP-complete, although there exist a number of useful approximation algorithms.

3.2 Reconstructing the Substitution Table

One possible way of determining S is to reverse engineer a *Nagravision* decoder and read the entire table out of its non-volatile memory. Since this procedure might be illegal in some regions, alternative approaches might be

attempted. A logic analyzer can be used to just observe the sequence of accesses to the line buffers B_i, which results in a large recorded collection of functions $v \in V_S$.

If we assume that opening the decoder is also not acceptable for legal reasons, we can use a PC video adapter to perform a chosen cipher image attack in which we send to the decoder a test image that contains genuine encrypted control word information in the vertical blank interval and that uses a redundant binary code to mark every line with its field line number. We record the descrambled test image and by reading the sequence of line number markers in there, we get the permutation p. If no access to a *Nagravision* decoder at all is allowed or possible, we can also attempt to use one of the Traveling Salesman approximation algorithms to determine samples of p from the correlation matrix of scrambled TV images alone as described in the previous section.

In both cases, we have to transform the observed permutations p into buffer access functions v before we can extract S. This can be accomplished with the following simple algorithm, provided that the given p is error-free: We set $b_i := i - 32$ for all $0 \le i \le 31$. Then for each line $0 \le j \le 254$ that the decoder outputs, we find the i for which $b_i = p(j)$ and we set both $v(j) := i$ and $b_i := j$. As a final check, we verify that after these 255 steps we have $b_i = p(255 + i)$ for all $0 \le i \le 31$.

In this way, we collect a number of members of V_S. Any of these reconstructed functions $v(i) = S((r + ti) \bmod 256)$ for $0 \le i \le 254$ shows all values of S except for one, but permuted by unknown parameters r and t. We just pick one v and chose our reconstructed table $\tilde{S}$ such that $\tilde{S}(i) := v(i)$ for all $0 \le i \le 254$. We then reconstruct another buffer access function v' and search for parameters a and b such that $v'((a + bi) \bmod 256) = \tilde{S}(i)$ for $0 \le i \le 254$ and once we found these (assuming we didn't by bad luck get some with $(a + b \cdot 255) \bmod 256 = 255$), we have also found the remaining value of $\tilde{S}$ with $\tilde{S}(255) = v'((a + b \cdot 255) \bmod 256)$.

We are not concerned about the fact that our $\tilde{S}$ is just a permuted version of S, because if $\tilde{S}(i) = S((a+bi) \bmod 256)$ for some (a, b) then this means that in the correlation search for the correct parameters (r, t) that follows now, we just find instead parameters $(\tilde{r}, \tilde{t})$ that compensate exactly this permutation of $\tilde{S}$ and result in the same v that we would have obtained with the correct table S and parameters (r, t).

3.3 Realtime Determination of the Permutation Based on a Known Substitution Table

3.3.1 Using Pixel Correlations

Once we know S either by extracting it from a *Nagravision* decoder or by determining it as outlined in the previous section, we can reverse the scrambling rather efficiently. A simple approach as implemented for instance in [12] is to perform a brute force search over all 2^{15} possible (r, t) tuples. For every possible (r, t) pair, the value of a penalty function is estimated by measuring the difference $|C_{x,p(y)} - C_{x,p(y+1)}|$ between a small number of randomly selected selected pixel pairs in the scrambled image that would under the tested (r, t) become neighbor pixels in the descrambled image. This can be implemented very efficiently since the permutation has to be performed only for the few test pixels and not for the entire image. We search for the (r, t) pair, for which the penalty function

$$H = \sum_{i=1}^{n} |C_{x_i,p(y_i)} - C_{x_i,p(y_i+1)}|$$

is minimal. The $(p(y_i), p(y_i + 1))$ pairs are precalculated for all 2^{15} (r, t) pairs for increased efficiency. Once this (r, t) pair has been identified, the corresponding permutation function is used to rearrange all lines in realtime.

A potentially much more efficient approach could be a binary subdivision search instead of a linear search over all 2^{15} possible (r, t) tuples. To implement this, we need a preparatory phase in which for a given substitution table S we build a binary decision tree. Each node in this decision tree lists a number of test pixel coordinates (x_i, y_i) and we branch to the left or the right subtree depending on whether H for these test pixels is above or below a threshold. Each leave of this tree is labeled with the (r, t) tuple that shall be used as the most likely candidate. A carefully built decision tree should be roughly balanced such that the maximum depth is not much over 15.

3.3.2 Using the SECAM Color Carrier

For *Nagravision* scrambled SECAM signals, there exists a simple alternative to looking at pixel luminosity correlations. In SECAM, color is encoded on a frequency modulated carrier in form of the two difference signals $R - Y$ (red minus luminance) and $B - Y$ (blue minus luminance) [8, 9]. The modulated $R - Y$ and $B - Y$ signals are added on alternating lines, where $R - Y$ uses

a 4.406 MHz $= 282 \cdot 15.625$ kHz carrier and $B-Y$ uses a 4.250 MHz $= 272 \cdot$ 15.625 kHz carrier to allow the TV receiver to synchronize its color decoder. The unmodulated color carrier signal is present on the front and back porch in the horizontal blanking interval and since it is permuted together with the active line, it is easy to see whether a scrambled line is in the descrambled field an odd or even numbered line. The sequence of 4.406 MHz and 4.250 MHz color carrier frequencies in the back porch of a scrambled SECAM signal is characteristic for the (r, t) pair that has been used to scramble this field. A pirate decoder only has to form a bit string representing the sequence of carrier frequencies found in the back porches of the lines 255–286 and use this bit string as the key in a hash-table lookup to access the (r, t) pair that descrambles this sequence correctly into one with alternating carrier frequencies. This (r, t) pair is then used to descramble the remaining field correctly.

Commercial hardware implementations of this attack became available in France around 1995 [13]. As a counter measure, the broadcaster *Canal+* uploaded a new substitution table S that ensures that the sequence of the color carrier frequencies is always alternating and therefore does not leak information about the (r, t) pair. An improved version of the attack looks not only at the frequency but also at the phase of the color carrier in the back porch. The SECAM color carrier is phase shifted by 180° for every third line to suppress visible dot patterns caused by the carrier signal. Again, a bit sequence that indicates which of the first 32 lines shows this phase shift acts as a hash-table lookup key to find the appropriate (r, t) pair quickly.

Since the PAL color burst is not permuted, the SECAM color carrier attack technique cannot directly be transferred to *Nagravision* for PAL. However, with the continuing introduction of the EBU Wide Screen Signal (WSS) in frame line 23 [14], the first line in every second clear field often has an easily recognizeable known structure. If the WSS line can be located in line w of the scrambled field, then we know that $0 > p(0) = w$ and $v(0) = w + 32 = S(r)$, which reduces the number of possible r values from 256 down to around 8 and speeds up the (r, t) search by a factor of 32.

4 Properties of the Substitution Table

The substitution table S used by the broadcasters *Premiere*, *Teleclub* and many others until today has the form

```
10, 11, 12, 13, 16, 17, 18, 19, 13, 14, 15, 16,  0,  1,  2,  3,
```

```
21, 22, 23, 24, 18, 19, 20, 21, 23, 24, 25, 26, 26, 27, 28, 29,
19, 20, 21, 22, 11, 12, 13, 14, 28, 29, 30, 31,  4,  5,  6,  7,
22, 23, 24, 25,  5,  6,  7,  8, 31,  0,  1,  2, 27, 28, 29, 30,
 3,  4,  5,  6,  8,  9, 10, 11, 14, 15, 16, 17, 25, 26, 27, 28,
15, 16, 17, 18,  7,  8,  9, 10, 17, 18, 19, 20, 29, 30, 31,  0,
24, 25, 26, 27, 20, 21, 22, 23,  1,  2,  3,  4,  6,  7,  8,  9,
12, 13, 14, 15,  9, 10, 11, 12,  2,  3,  4,  5, 30, 31,  0,  1,
24, 25, 26, 27,  2,  3,  4,  5, 31,  0,  1,  2,  7,  8,  9, 10,
13, 14, 15, 16, 26, 27, 28, 29, 14, 15, 16, 17, 18, 19, 20, 21,
22, 23, 24, 25,  5,  6,  7,  8, 19, 20, 21, 22, 12, 13, 14, 15,
17, 18, 19, 20, 27, 28, 29, 30, 10, 11, 12, 13, 11, 12, 13, 14,
 6,  7,  8,  9,  1,  2,  3,  4,  0,  1,  2,  3,  4,  5,  6,  7,
 3,  4,  5,  6,  8,  9, 10, 11, 15, 16, 17, 18, 23, 24, 25, 26,
29, 30, 31,  0, 25, 26, 27, 28,  9, 10, 11, 12, 21, 22, 23, 24,
20, 21, 22, 23, 30, 31,  0,  1, 16, 17, 18, 19, 28, 29, 30, 31
```

This particular table has the curious property

$$S(i) = (S(i - i \bmod 4) + i \bmod 4) \bmod 32,$$

but it is not clear what the reason behind this is.

The broadcaster *Canal+* in France replaced the above table in September 1997 with the following one, as a response to the availability of unauthorized *Nagravision* for SECAM decoders that reconstructed the (r, t) pair from the sequence of the color carrier frequencies in the scrambled image:

```
 0,  1,  2,  3,  4,  5,  6,  7,  2,  5,  4,  7,  8,  9, 10, 11,
14, 17, 16, 19, 22, 25, 24, 27, 28, 31, 30,  1, 24, 27, 26, 29,
 8, 11, 10, 13, 20, 23, 22, 25, 20, 21, 22, 23, 30, 31,  0,  1,
16, 17, 18, 19, 28, 29, 30, 31, 10, 11, 12, 13, 16, 17, 18, 19,
12, 15, 14, 17,  0,  1,  2,  3, 20, 23, 22, 25, 18, 19, 20, 21,
22, 25, 24, 27, 26, 27, 28, 29, 18, 21, 20, 23, 10, 13, 12, 15,
28, 29, 30, 31,  4,  5,  6,  7, 22, 23, 24, 25,  4,  7,  6,  9,
30,  1,  0,  3, 26, 29, 28, 31,  2,  5,  4,  7,  8,  9, 10, 11,
14, 15, 16, 17, 24, 27, 26, 29, 14, 17, 16, 19,  6,  9,  8, 11,
16, 19, 18, 21, 28, 31, 30,  1, 24, 25, 26, 27, 20, 21, 22, 23,
 0,  3,  2,  5,  6,  7,  8,  9, 12, 13, 14, 15,  8, 11, 10, 13,
 2,  3,  4,  5, 30, 31,  0,  1, 24, 25, 26, 27,  2,  3,  4,  5,
30,  1,  0,  3,  6,  9,  8, 11, 12, 15, 14, 17, 26, 27, 28, 29,
14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25,  4,  7,  6,  9,
18, 21, 20, 23, 12, 13, 14, 15, 16, 19, 18, 21, 26, 29, 28, 31,
10, 11, 12, 13, 10, 13, 12, 15,  6,  7,  8,  9,  0,  3,  2,  5
```

This new table has the property $S(i) \equiv i \pmod 2$. This way, the resulting permutation always alternates between odd and even numbered lines and the sequence of color subcarrier frequencies in a scrambled SECAM image can only reveal the value $r \bmod 2$ and not the complete r and t values.

If S were selected such that

$$i \not\equiv j \pmod 6 \quad \Longrightarrow \quad S(i) \neq S(j),$$

then even the combination of color carrier phase and frequency, which is repeated every six lines in the clear signal, could only reveal $r \bmod 6$ and $t \bmod 3$. Apparently the *Canal+* or *Kudelski* technicians who designed the above countermeasure table failed to understand the threat of a color carrier phase analysis at that time.

5 Conclusion and Final Remarks

Nagravision uses a surprisingly weak scrambling technique that can rather easily be defeated without using any cryptographic secrets that might be stored in the subscriber smartcard. While image processing attacks can only approximate the original signal for cryptographic scrambling systems such as *VideoCrypt* and *EuroCrypt*, *Nagravision* allows the attacker to determine reliably the seed value in such a short time that the clear image can be reconstructed without any quality loss in real time using standard personal computers or decoder designs that cost not much more than the official decoder. The color-carrier sensing pirate decoders for the SECAM version of *Nagravision* can easily be defeated by a more carefully designed substitution table. Whether lasting countermeasures are possible against pixel-correlation based pirate decoders depends on whether the broadcasters can upgrade the fielded decoders easily to use a larger set of permutation parameters than 2^{15} and whether v can be replaced by a cryptographically strong cipher function.

This paper is work in progress and might still contain errors. I started writing it in order to get a better understanding of the mathematical properties of the *Nagravision* scrambling method and the algorithms used in the various currently available pirate decoders. These have been designed by individuals who want to stay anonymous because they are afraid that the work on these decoders might be considered illegal in their home country (France). I also wrote this paper to collect and discuss possibly useful ideas and insights towards more advanced attacks and countermeasures. Since the *Nagravision* system is anyway scheduled to be replaced by a DVB conditional access

system, I do not think that publishing my thoughts on the topic can do any economic harm, but I hope they might be of some educational benefit.

Special thanks to Fabian Petitcolas, Roberto Deza Asensio, and "Zorglub" for comments on the paper and to Ralph Metzler for providing frame-grabber images for experiments. Suggestions for improving this text are very welcome.

References

[1] Andre Kudelski. Method for scrambling and unscrambling a video signal. United States Patent 5375168, 20 December 1994.

[2] Access control system for the MAC/packet family: EUROCRYPT. European Standard EN 50094, CENELEC, December 1992.

[3] Vincent Lenoir. EUROCRYPT, a successful conditional access system. *IEEE Transactions on Consumer Electronics*, 37(3):432–436, August 1991.

[4] Michael Cohen and Jonathan Hashkes. A system for controlling access to broadcast transmissions. European Patent Application 0 428 252 A2, 22 May 1991.

[5] Ross J. Anderson and Markus G. Kuhn. Tamper resistance – a cautionary note. In *The Second USENIX Workshop on Electronic Commerce Proceedings*, pages 1–11, Oakland, California, 18–21 November 1996.

[6] V. Mangulis. Security of a popular scrambling scheme for TV pictures. *RCA Review*, 41(3):423–432, September 1980.

[7] D. Raychaudhuri and L. Schiff. Unauthorized descrambling of a random line inversion scrambled TV signal. *IEEE Transactions on Communications*, 31(6):816–821, June 1983.

[8] Television systems. ITU-R Recommendation BT.470, International Telecommunication Union, Geneva.

[9] Jim Slater. *Modern Television Systems – to HDTV and beyond*. Pitman, London, 1991.

[10] Thomas H. Cormen, Charles E. Leiserson, and Ronald L. Rivest. *Introduction to Algorithms*. MIT Press, 1990.

[11] Eugene L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys. *The Traveling Salesman Problem*. John Wiley & Sons, 1985.

[12] "Gaston". NagraTV 2.0. Internet `http://eurosat.com/salp/gaston/`, June 1998. Linux open source software in C.

[13] "Zorglub". La page technique du Nagravision [The Nagravision technology page]. Internet `http://eurosat.com/salp/zorglub/`, 1998. (in French).

[14] Television systems; 625-line television Wide Screen Signaling (WSS). ETS 300 294, ETSI, Sophia Antipolis, September 1997.

[15] John McCormac. *European Scrambling Systems 5 – The Black Book*. Waterford University Press, 1996. ISBN 1-873556-22-5.

[16] V. Zacharopouloulos, Ch. Mantakas, K. Dagakis, and C. Caroubalos. An analogue scrambling scheme for television signals. *International Journal of Electronics*, 59(4):501–509, October 1985.