A framework for secure MIDI eCommerce

Jana Dittmann - Martin Steinebach[1]

Electronic commerce has gained importance over the last years, but only a few digital products can be sold and delivered over the internet. These products need security to be protected against pirates who could sell copies over the internet without any loss of quality. MIDI files are an excellent example for such a product: They are rather small and expensive, making them ideal for downloading over the net, but also for building up large illegal collections on pirate web sites or CDs. We discuss a security framework for MIDI files protecting the complete transaction between web shop and client and show how MIDI files can be protected even after downloading by embedding digital watermarks to prevent illegal reselling or distribution.

1. Motivation and introduction

In this paper we address the problems selling MIDI (Musical Instrument Digital Interface) files over the internet. A file size of usually under 100 kb and a price of 10 Euro and more make them well suited for online commerce. There are numerous online stores offering a wide range of MIDI files for musicians. These attributes also make MIDI files attractive for pirates. Trading or illegal distribution is common.

Therefore methods for securing the property of the rightful owners are necessary. In chapter 2 we discuss a framework for secure MIDI eCommerce using digital watermarking and encryption. Unlike other multi media data, distributing encrypted MIDI files is possible without major drawbacks: On the one hand the small file size allows fast on demand encryption. On the other hand an error while transmitting the file makes it useless, encrypted or not, as a MIDI file is more like a program than like a multimedia stream, the loss of several notes will not be accepted by a client. But encryption can only secure transmission. The MIDI file has to be decoded to be used in a sequencer and could be copied and transmitted without protection thereafter. Proprietary file

formats are also useless, as MIDI information transmitted to synthesizers can be recorded with the help of computers or hardware sequencers. Watermarking is a possible solution to secure MIDI data beyond transmission. In chapter 3 we show different approaches to embed information directly into the MIDI data. This protection is transmitted and recorded again with every copy.

1.1. MIDI

MIDI was invented as an standard for accessing musical instruments via remote keyboards. It consists of a set of instructions like playing a note, changing the volume or a sound bank and a protocol how these instructions are sent to the instrument [1], [4]. The standard includes 16 different channels to address different instruments over one MIDI network and a set of controllers for changing pitch, volume, vibrato and other musical information. *Figure 1* illustrates the basic structure of MIDI data. MIDI also offers a way to send specific data for an instrument (e.g. a sound patch) called sysex. The file format of MIDI is called Simple MIDI File (SMF). There are at least three versions (SMF0, SMF1, SMF2) which differ only in the way how the data is arranged inside of the file and have no consequences on the information discussed in this paper. There are also some additional standards for instrument to MIDI sound bank mapping, parameter standardization, effect algorithms and minimum MIDI instrument capabilities called GM (Gerneral MIDI) and Yamahas XG.

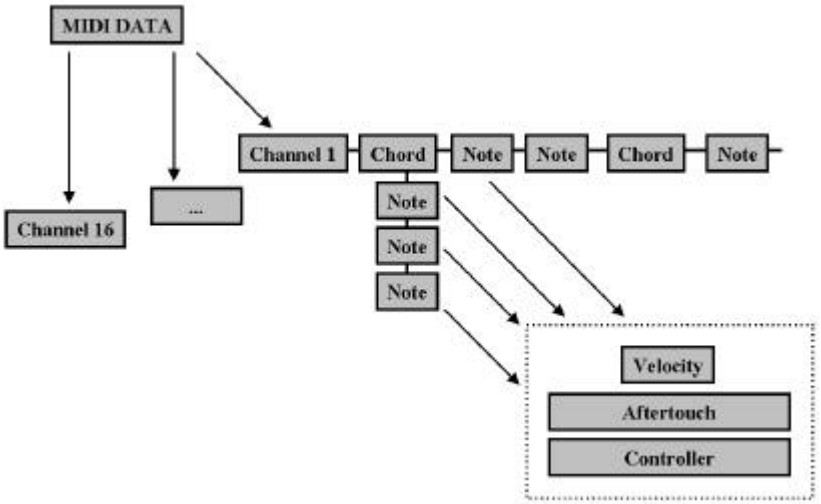

Figure 1 : Basic MIDI data structure

---

[1] GMD - German National Research Center for Information Technology, Darmstadt, Germany
{jana.dittmann;martin.steinebach}@darmstadt.gmd.de

1.2. Digital Watermarking

Digital Watermarking is a technology capable of solving important practical security problems. It is a highly multidisciplinary field that combines media and signal processing with cryptography, communication theory, coding theory, signal compression, and the theory of human perception. By digital watermarking data is embedded within another data stream or signal using aspects of the carrier signal like quantisation noise in A/D-conversion. Interest in this field has recently increased because of the wide spectrum of applications it addresses. In [2] in five different application types is given: authentication or copyright watermark, fingerprint watermark, copy control or broadcast watermark, annotation watermark and integrity watermark. The most important properties of digital watermarking techniques are robustness, security, imperceptibility/ transparency, complexity, capacity and possibility of verification.

2. MIDI eCommerce

MIDI is an excellent example of a digital product which can be sold with an eCommerce solution. As delivery can be done electronically with FTP or email, there is no break in the chain of online solutions for MIDI commerce. We will show how a virtual shop for MIDI files could look like, which possibilities there are and what makes it more convenient and fitting to the medium then a usual store.

Customers of MIDI files often use them on professional basis. Either they are musicians playing back the files on keyboards or computers or they are teachers using the files as examples. In most cases one of the most important needs is to be able to use MIDI files of new songs from recent music charts. So a framework which provides a fast and convenient way to obtain the files is very desirable. An example: A musician is hired to play a birthday party. He is asked to play the favorite song of the host at the evening, a song from the Top 10, only a few weeks old. Without an online store, it would hardly be possible for the musician to get the song until the evening.

There are at least three parties involved in MIDI eCommerce (*figure 2*): The one who creates the score, the web shop and the musician who buys it. Additional parties could be the artist or label owning the copyright of the song, some instance controlling copyright issues (like GEMA in Germany) and a bank providing a way to pay online. Online payment is necessary for the whole transaction to happen on the internet. Without it, invoices have to be sent to the identified client slowing down the process. As eCommerce started, using credit cards for online payment was

common, but may users did not trust internet security. Therefore standards for online payment like SET (Secure Electronic Transaction), providing a way to facilitate secure payment card transactions over the internet. Electronic cash is another payment method designed for online payment. Cybercash, DigiCash and VisaCash are examples for payment systems similar to real world cash.

This paper concentrates on security between the web shop and the client.

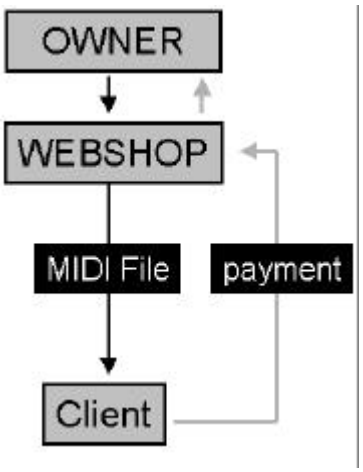

Figure 2: The three parties involved in a simple MIDI file transaction

2.1 Secure MIDI eCommerce

Using the watermarking approaches described later in chapter 3 and algorithms for encryption, we will introduce a framework in which a MIDI file can securely be sold to a customer. Watermarking is used to track clients who pass the MIDI files on to third parties. Every MIDI file is stamped with the client ID using a watermarking algorithm. If a MIDI file is found somewhere in the internet or on a computer, the original client can be identified by reading the watermark.

Many MIDI web shops deliver the MIDI files by email. If these emails are not secured, a client could claim that the file had been pirated before he got it. Here cryptography can be used to secure the file until it reaches the client. If the webshop sends an encrypted file using a secret key only known to the client, only the client will be able to decode and use it. Every copy of the file marked with his ID that is found must have been made after he decrypted it. An alternative to the secret key scheme is using the clients public key provided by a PKI. Only the client can decrypt the file using his secret key.

2.1.1. Encryption

The watermarked MIDI file can only be sent to the client if it is certain that no third party can capture and use it. Therefor it is encrypted before the transmission. If it is captured, a third party can not use it without a key for decrytpion. A common method for encrypting multimedia files is a combination of symmetric and asymmetric encryption methods. The file is encrypted using a fast symmetric algorithm and a random key K. K is encrypted using an asymmetric method with the clients public key. Both encrypted files are sent to the client who first decrypts K with his private key and then decrypts the MIDI file with K. This method is faster then to encrypt the whole MIDI file with the public key and therefor saves resources on the web server.

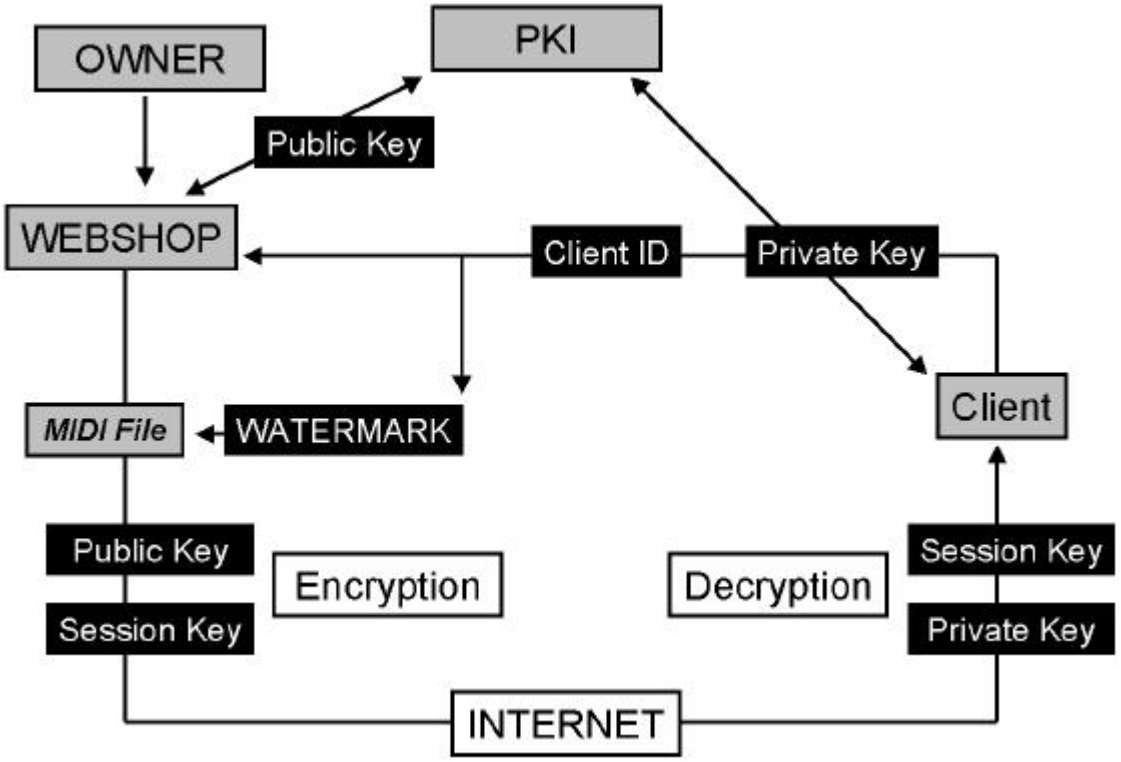

Figure 3 : Framework structure for a secure MIDI file transmission via internet

2.2. A secure MIDI eCommerce framework

As stated above, we will only discuss the security of the MIDI file transaction between web shop and client. *Figure 3* illustrates such a framework using the security methods mentioned in 2.1.

The client chooses to buy a MIDI file. He identifies himself at the web shop sending his client ID and some identification message encrypted using his secret key to prove his identity. The web shop requests the users public key over some public key infrastructure (PKI). If the message is decoded without errors, the web shop will send the MIDI file. It embeds a watermark with the client ID into the MIDI file. Then the file is encrypted using a session key which is encrypted using the clients public key. Both file are given to the client via internet by download or email. The client uses his private key to decrypt the session key and then decrypts the MIDI file. He can use it now, but the file is still marked with his user ID. So if it is found decrypted somewhere on the web, the client will be identified and legal means will be taken.

3. Possible methods for MIDI-Watermarking

Watermarking MIDI files is different to most of the published multimedia watermarking methods. Usually, there is an analog original which is digitized and watermarked. The digital representation is as close to the digital as required for the intended application. As digitizing images or sounds produces noise to the signal, this noise can be used to hide the watermark. Usually there is more information in a digital sample then a human can perceive with his senses. This provides another possibility for hiding a watermark: at the edge of perception.
MIDI data is more similar to text or software watermarking: There is no noise, as MIDI is only a description of the musical piece, a score with some added features. To embed a watermark in a MIDI file, the existing information has to be changed carefully, as random manipulation would lead to an useless file as instructions could be corrupted.

3.1.Timing modulation

MIDI offers a very high timing resolution if required. Every information in a SMF includes a delta-time to ensure proper timing of the included musical events. Using a very high resolution not necessary for the musical score a watermark can be embedded by modulating the delta time amount of certain events. If, for example, a bass motive that consists of quarter notes of equal delta time distance has its delta time changed, the timing of the quarter notes is not exactly on the beginning of each quarter, but has a small offset which is used as watermarking information. This method will be transparent if only very small amounts of timing are changed. A quantization attack will remove the watermark. But it also destroys the individual groove of the musical piece, because this groove

consists of similar timing irregularities produced by human players. Another attack is the "humanize" option offered by many sequencers. Here small timing changes are applied on the computer generated score to give it a human touch, as it was played on a keyboard and recorded via MIDI. This changes the perceived musical information, therefore this attack will not be applicable on high quality MIDI files.

3.2. Chord saturation

Most musical pieces use chords. An algorithm could find these chords and add notes to it. Thereby chords can be used as information carriers. Transparency is certainly endangered in this method, and most composers would not allow such changes to their score. But if only lower notes of small attack velocity are added, the rest of the chord could mask the change for the average listener. Attacks against this watermarks are only possible if the attacker has a certain understanding of the marked musical piece, or else he will delete the wrong notes and thereby change the perceived musical piece.

3.3.Velocity patterns

A velocity number is given with every note transmitted via the MIDI system. It describes, how fast the key on the keyboard belonging to the note has been pressed. The faster one presses the key, the louder the note will be played. The range of it is 0 to 127. As every sound device interprets the velocity individually, slight differences in the interpretation occur. Therefore it can be assumed that an algorithm can be applied to change the velocity numbers by small amounts to create patterns to embed a watermark (*figure 4*). This method is similar to the MPEG scale factor watermark we introduced in [3]. To embed information, bits are assigned to patterns. The patterns are redundantly embedded into the MIDI file. To retrieve the watermark, a detector has to search for these pattern in the MIDI file and to decode the binary information belonging to the found patterns.

There are a number of special cases to be treated. Some synthesizers use a threshold in velocity for certain sound changes. At a discrete step, the sound changes from normal to distorted. If the algorithm changes numbers close to this threshold, transparency is possibly lost. Some other synthesizers and samplers use stacking. Here different sounds are placed on the same key and are played dependably of the velocity number. Changes in the number could mean changes in the choice of sound. These problems are special cases, and most distributed MIDI scores will not

address special instruments but workstations, where such sound changes usually do not occur. To further reduce the probability of transparency loss, only velocity numbers within certain ranges should be changed, a reasonable range would be 20 to 100. Possible attacks to this method are velocity quantization attacks which would also damage the perceived dynamics for a listener. Filtering, re-recording or changes in the MIDI format will have no influence to the velocity numbers as it can be assumed that a total loss of dynamics will not be accepted by the listener.

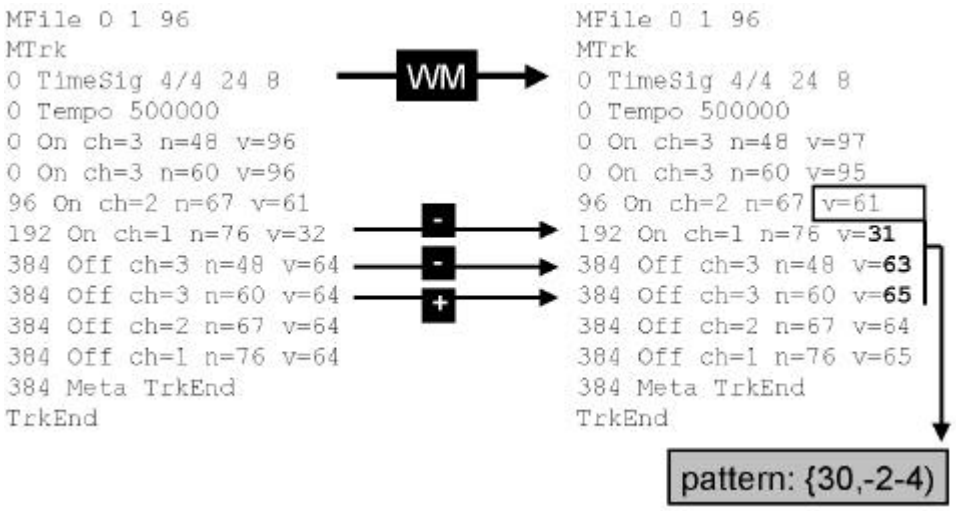

Figure 4 : Velocity pattern watermarking

3.4. Attacks

When watermarking is used to provide security, there is always the possibility of attacks against the watermarks. In the case of MIDI, a typical attacker will either use a commercial sequencer like Cubase or Logic or utilities designed for MIDI file manipulations.

Sequencers will offer attack possibilities like changing channel numbers, adding or splitting of channels, random note manipulation, for example velocity, length or aftertouch, quantization or "humanization" of the whole score or sequences, timing offsets and filtering sysex information. Utilities can offer additional attack possibilities like MIDI to text conversion to manually search for watermarking information or statistical attacks to find information patterns. Using a sequencer is easy. Everyone interested in MIDI will be able to apply basic attack manipulations with it. Attacks with utilities are also easy, but not every user will know how to use it or even have access to such software.

Therefore a MIDI watermarking solution should mainly address attacks by sequencers. These attacks and some estimations about robustness have been discussed in the paragraphs above. Robustness tests will be done after the described methods are implemented.

4. Summary

MIDI files have been introduced as a excellent example for an online commerce product. We have shown a framework for secure MIDI eCommerce using encryption, watermarking and a PKI. Concepts for MIDI watermarking have been discussed, providing different ways to embed information into the MIDI files to track illegal copies back to the source. These concepts include embedding patterns in velocity information or timing and changing chords. MIDI has been shown to be different then most multimedia data, as it is more like a music programming language then like recorded music.

Attackers who want to remove the watermark will either use sequencers with different MIDI editing functions or specialized tools utilizing statistical analysis. The robustness of the different watermarks against this attacks will be high enough to prevent the removal of the watermark without quality loss.

4.1. Future work

The concepts of MIDI watermarking have to be implemented to evaluate transparency, robustness and complexity. Without an implementation we can only make assumptions about these important parameters. As MIDI does not sound the same on each MIDI device, even f using standards like General MIDI, the transparency tests will have to be done on different devices to prove inaudibility of the watermarks.

Additional MIDI watermarking concepts are possible. Fingerprinting could be used to identify groups of pirates using more then one copy of a MIDI file for a collision attack. Content-fragile watermarking could detect changes in the musical information to secure the quality of a MIDI file. Both concepts are described for image in [2].

References

[1] BUICK, P.; LENNARD, V., Music Technology Referenece Book, PC Publishing, 1995
[2] DITTMANN, J., Digitale Wasserzeichen, Springer Verlag, ISBN 3 - 540 - 66661 - 3, 2000

[3] DITTMANN, J., STEINEBACH, M., STEINMETZ, R., Digital Watermarking for MPEG Audio Layer 2, Proceedings of ACM Multimedia'99
[4] RUMSEY, F., MIDI Systems & Control, Focal Press, 1994