Inhibiting Card Sharing Attacks

Michael Tunstall, Konstantinos Markantonakis, and Keith Mayes

Smart Card Centre, Information Security Group,
Royal Holloway, University of London,
Egham, Surrey TW20 0EX, UK.
{m.j.tunstall, k.markantonakis, keith.mayes}@rhul.ac.uk

Abstract. The satellite TV industry relies heavily on the use of smart card technology at the very heart of broadcasted services that are protected by legacy conditional access systems. The process of Satellite TV signal protection is distributed amongst a number of system components, e.g. smart cards, receivers, Conditional Access Modules (CAM) and the content provider. However, the introduction of "Open" Satellite Receivers, providing a highly configurable environment with software emulation of conditional access systems, enabled the implementation of whole range of new attacks. A widely deployed attack is often referred to as the "card sharing" attack, by which one legitimate user colludes to provide protected content to a larger group of unauthorised users. This paper proposes a countermeasure that increases the bandwidth requirements of this attack to the point where it is no longer practical with a standard internet connection, with a minimal impact on existing protocols and architectures.

1 Introduction

During the early development of the satellite TV industry it became evident that in order to protect its investment and revenue streams it was necessary to encrypt digital content. Protection of the digital content traditionally relied on a number of system components including Set-Top-Boxes (STB), smart cards and content encryption boxes at the service provider level. Encryption, in the context of the satellite TV industry, is often defined as "the process of protecting the secret keys that have to be transmitted with the scrambled signal in order for the descrambler to work". The above procedure requires the existence of a conditional access system [19, 13] that combines a signal encryption algorithm and key protection algorithm in order to prevent unauthorised signal reception. Many providers follow the DVB-S standard [9] and tailor the necessary configuration parameters [11] to their own particular needs.

The recent technological advances of the computer industry, along with the continued requirements for more advanced and powerful set of services means that satellite TV providers are all trying to differentiate their offerings, each with their own STB software and hardware. Conversely, consumers are constantly looking for ways that will allow them to use more flexible and powerful equipment that will simplify or even enhance their viewing experience.

The natural market response was the introduction of open satellite receivers [30] that allowed consumers to purchase the STB that met their viewing and personal requirements. These STBs are highly configurable environments based on open operating systems such as Linux. Basically, they can be considered as traditional computer workstations enhanced with satellite TV signal processing capabilities. However, the fact that they are powerful and open devices has introduced a number of new threats. At the very least, it has enabled a more efficient realisation of a number of already identified and well-documented attacks [20, 22, 16]. The market trend towards open receivers along with current hardware restrictions (e.g. at the card level) has forced the satellite TV content providers to mitigate their protection mechanisms away from the underlying STB hardware and bring it closer to the smart card and Conditional Access Module (CAM). The latter two are often seen as a single component.

The purpose of this paper is to briefly introduce how these recent technological advances are affecting the satellite TV industry and then to propose a relevant countermeasure for a specific type of attack i.e. the card sharing attack [22].

The aforementioned open satellite receivers can be used to share one subscriber's rights with numerous people. A subscriber can start a server on a STB that will accept connections from other open STBs. These client STBs belong to people that are not subscribers and therefore have no rights to view a given broadcast. Every time the client STB receives an encrypted key from the broadcaster, it is sent to the server STB that deciphers it and returns the key necessary to decipher the broadcast. As the maximum frequency that this key can be changed is once every two seconds, the amount of bandwidth required for this is negligible (a maximum of 5 bytes of information per second in each direction). With the bandwidth requirement being so small synchronisation issues can be minimised.

Furthermore, it implies that one receiver can act as a server and provide numerous clients with the sequence of keys necessary to watch a given broadcast. As an open receiver can simulate a CAM it is difficult to base a solution at this level as any behaviour can be simulated. The CAM should therefore be regarded as an untrustworthy entity within the protocol. This paper will propose a smart card based solution to this sharing attack that will mean that all users watching a given broadcast will need to have a smart card issued by the broadcaster.

The remainder of the paper is structured as follows: Section 2 provides all the necessary background information that will enable the reader to understand how content encryption works in the satellite TV industry along with the recent smart card attacks in the light of the introduction of open STBs. In Section 3 we provide an overview of the proposed countermeasures and how different communication protocols change the countermeasure. Section 4 discusses how the proposed countermeasure affects users with various different types of countermeasure. This is followed by the conclusion in Section 5.

2 Issues around Content Provision for the Satellite TV Industry

In the following sections we provide an overview of how the situation is changing in the light of the recent technological advances in the smart card, STB and satellite TV industries. Subsequently, the main characteristics of two widely used satellite TV attacks are highlighted in order to provide a reference point for the proposed countermeasure.

2.1 A Changing World – New Requirements for Open Receivers

An Open Receiver (OR) or Open Set-top-Box (OSTB) is a highly reconfigurable computer system that offers the capability to receive and decrypt the scrambled TV signals. These receivers often come with a number of pre-installed Conditional Access Module (CAMs) along with a Linux operating system. Furthermore, they also come pre-installed with a number of "images" containing all the necessary software to watch subscribed TV channels, along with various other tools for recording and organising channels using a variety of graphical interfaces.

However, the hacking communities are taking advantage of these open receivers by developing their own "images" containing all the necessary hacking tools that will enable them to circumvent the security around a protected TV signal. These images reside in the EEPROM or flash memory of the OSTB and can be easily upgraded, deleted and modified by connecting the receiver to another computer through a network or serial cable. A variety of the plug-in images enable the receivers to access USB tokens, hard disks, connected cameras or keyboards and to use the network or modem cards. All this functionality along with the plethora of freeware hacking tools makes the open receiver a very powerful tool in the hands of illegitimate users.

2.2 Recent Satellite TV Attacks

Over the last decade a number of satellite TV attacks have emerged. Some of them are based on cards being cloned, communication being logged and on-card elements being emulated by software residing in the STB. In the following paragraphs we provide a very brief overview of the main type of attack that has particular significance in the light of the proposed countermeasure.

The Card Sharing attack [20], see Figure 1, belongs in a set of simple, powerful and effective satellite TV attacks. This attack requires an OSTB with a legitimate card (i.e. the Server CAM), sharing its secrets with a number of illegitimate receivers (i.e. Client CAM) in order to provide them with access to unauthorised content. The user with the legitimate card runs a Card Server image on their OSTB.

The server image enables the OSTB server to accept connections from a number of OSTB clients (Client CAM) across a number of communication mechanisms including the Internet. As soon as an Electronic Management Message

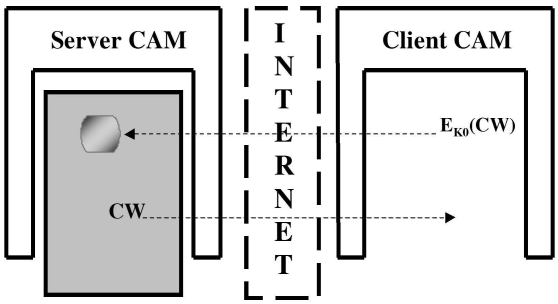

Fig. 1. Overview of the card sharing attack.

(EMM) or an Electronic Code word Message (ECM) is received by an OSTB client it is forwarded to the corresponding OSTB server and in turn to the corresponding smart card in order to be processed. The server subsequently carries out the message decryption and forwards back to each client the decrypted CW or other relevant messages. This type of attack is often referred to as the jugular attack [22].

2.3 The Concept of Secure Content Provision for the Satellite TV Industry

Satellite communication is considered a very expedient vehicle for broadcasting a large amount of valuable information over wide geographical areas. The satellite TV industry needs to rigorously safeguard its revenue streams, i.e. the content or "Services". Various sources [18, 7] estimate that in Australia, for example, approximately 4-5% of all satellite TV subscriptions were illegal resulting in a direct loss in the realm of 50 million Australian dollars. Broadcasters wanting to protect their revenue streams will therefore have to employ some effective and robust means to control access to the transmitted services.

The process of satellite TV signal transmission is often divided into two distinct phases: The first begins with the service provider encrypting or "scrambling" the signal and the second when the subscriber uses the necessary equipment, i.e. STB, in order to decrypt the signal. There are several systems that can provide access control for satellite TV; the most widely used ones are presented in [15, 21].

Digital Video Broadcast (DVB) is a broadcasting standard developed by the major European satellite TV producers. The DVB standard is based on the MPEG-2 standard [1] that organises broadcasts into packets separating multiplexed information from program streams. The most commonly deployed satellite TV broadcasting methods involves a STB, a satellite dish responsible for receiving the encrypted signal, a Conditional Access System (CAS) [6] which often includes a CAM and a smart card that is responsible for the service decryption.

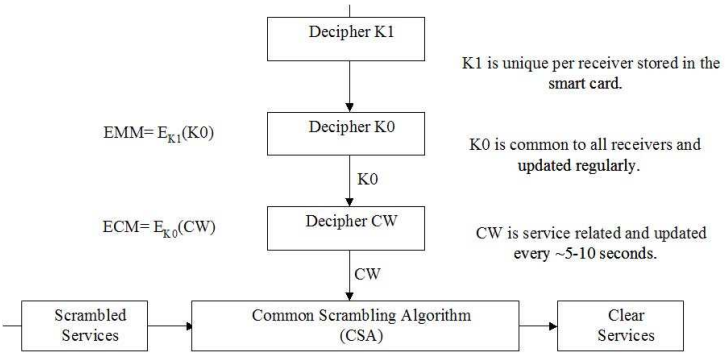

Fig. 2. Summarised process of DVB signal de-scrambling and cryptographic key hierarchy.

The process is simplified in Figure 2 and summarised as follows. The service signal is encrypted/scrambled (by using the DVB Common Scrambling Algorithm) using a cryptographic key, called a Code Word (CW), that is generated by a Control Word Generator.

In turn, the CW is encrypted and encapsulated within an ECM, in order to protect the CW during transmission to all legitimate recipients. The encryption of the CW is often defined as "the process of protecting the keys that will be transmitted with the scrambled signal in order for the descrambler to work". A CAS offers all the necessary flexibility to satellite TV operators to operate proprietary conditional access systems that better fit their security and operational requirements. Some of the most commonly used CAS systems include VIAaccess [29], Irdeto [17], Cryptoworks [8], and Seca [26]. In principle, the CAS prohibits brute force attacks from taking place as the signal encryption key is changing every 2–10 seconds [5], i.e. the crypto period. The details of the CAS remain confidential but the basic idea that a chain of encryptions is taking place on the CW to guarantee protection of keys and avoid brute force attacks.

The role of the STB is to receive the satellite TV signal through the satellite dish and return the descrambled stream. This actually involves the utilisation of both the smart card and the CAM. The multiplexed/scrambled services and ECM are forwarded to the CAM residing within the STB. The actual ECM is forwarded from the CAM to the smart card. A Service Key (K0) is stored in the smart card and it is used in order to decrypt the CW. An Electronic Management Message (EMM) updates these keys, and their validity period is usually one month but varies from one broadcaster to another. The newly obtained CW is also used within the CAM in order to decrypt the signal and return it back to the STB. The ECM and EMM can be used in order to send commands and new keys to the smart cards. In the above architecture the STB can host multiple CAMs in order to match the individual broadcaster requirements.

In terms of the DVB data broadcast the following process is incorporated: the DVB transmission is an encrypted signal that has a bandwidth of 1 to 4 Mbits per second in packets of 188 bytes. The encrypted signal is accompanied by series of control words (CW) that can be deciphered by receivers to provide a key (CK) that can be used to decipher the broadcast. The CW can be updated during a broadcast so that more than one key is needed to decipher the broadcast. The maximum frequency this can occur is once every two seconds [5]. The keys for transforming CW to CK from a set of 292 keys that are distributed and updated by the broadcaster. This key is then used to decipher the payload of the DVB packets being delivered to each STB.

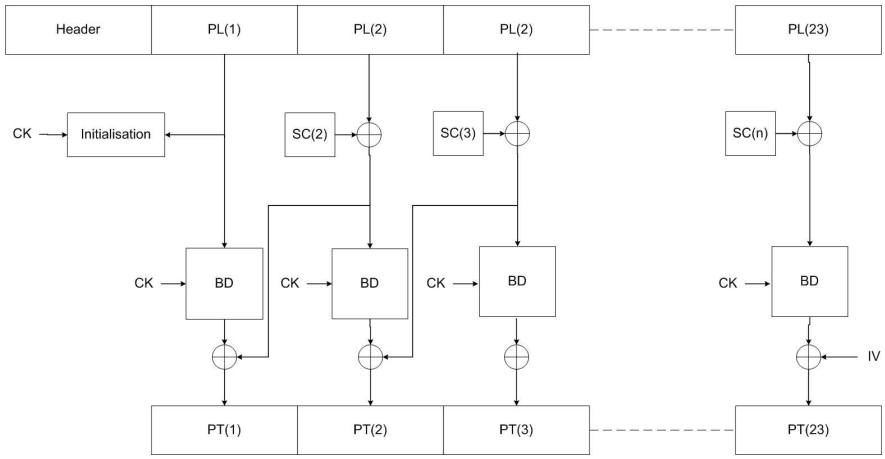

Fig. 3. Deciphering the contents of DVB packets using key CK.

Figure 3 shows the deciphering process that comprises of two layers. The payload (PL(i) for $i \in [0, 23]$) of the packet is first deciphered by a stream cipher (SC) and then deciphered by a block cipher (BD) using the CBC chaining method. This produces the plaintext of the signal (PT(i) for $i \in [0, 23]$). The value for i can take values in the interval $[0, 23]$ as the stream cipher and block cipher treat the data in blocks of eight bytes. Further information on this process is available at [2], although the authors admit that in an actual implementation the details may vary.

In most cases this process takes place in a controlled environment where the STB and CAM are provided by the broadcaster. If an OSTB is used the whole process can be simulated and each incoming CW is sent to a server OSTB that will return CK, that can then be used to decipher a given broadcast. Every time CW is changed the client OSTB is required to send and receive a message of 10 bytes. If CW is changed every 2 seconds (the maximum frequency) this gives a bandwidth of 40 bits per second in each direction.

3 Increasing the Bandwidth Requirements

A way of raising the difficulty of this attack would be to use the fact that a smart card contains all the keys necessary for deciphering the arriving CWs. It is assumed that these keys can be delivered securely and are not at risk once stored within the smart card.

The smart card could be used to create a stream of values for CK rather than one value that is valid for 2–10 seconds. As the deciphering key changes much more frequently a server OSTB would have to provide much more information to enable a client OSTB to decipher a broadcast. This would have two effects, the amount of bandwidth necessary to share viewing rights becomes prohibitively large, and as the smart card is constantly communicating (i.e. it's bandwidth is saturated) it can no longer be asked to decipher arbitrary CWs as it can only create one stream at a time.

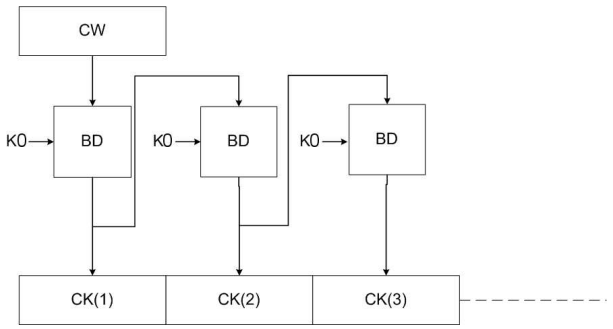

Fig. 4. Generating a series of values for CK.

For a given CW a smart card can deliver a series of CKs, as shown in Figure 4, that are used to decipher the signal until the delivery of another CW. The arriving CW is deciphered with a block cipher (BD) using the key (K0) to produce a CK, this is again deciphered to produced the next CK etc. The value for K is chosen from the 292 available by the header delivered with CW. This means that an attacker would be obliged to share each CK as it is generated to enable someone else to decipher the same signal. New values can be generated by continuing to decipher the delivered CW to produce other values of CK. When a new CW is delivered the process restarts with the new CK.

Two different possibilities for implementing this idea are discussed below: using a standard smart card and using a card with a fast protocol to provide the sequence of CK values.

3.1 Using a Standard Smart Card

In order to ask a smart card for a new value for CK the CAM will need to send an APDU and then receive the procedure byte, the data, and status word [23]. This gives an overhead of 8 bytes for each request to the card for a new value for CK. In order to minimise the effect of this, numerous values of CK can be delivered by the same APDU to minimise the amount of protocol bytes sent. These keys will be stored on the CAM and used as necessary. Table 1 gives the set of keys that can be generated per second for every amount of keys that can be delivered in 1 APDU. The bandwidth requirement is shown as a function of the number of CKs provided per APDU.

Table 1. The bandwidth requirements for different length commands.

CKs per APDU	Clock cycles per command	CKs per per second	Bandwidth required (Kbits/s)
1	3072	1627.6	104.2
2	4608	2170.1	138.9
3	6144	2441.4	156.2
4	7680	2604.1	166.7
5	9216	2712.6	173.6
6	10752	2790.1	178.6
7	12288	2848.3	182.3
8	13824	2893.5	185.2
9	15360	2929.6	187.5
10	16896	2959.2	189.4
11	18432	2983.9	191.0
12	19968	3004.8	192.3
13	21504	3022.6	193.5
14	23040	3038.1	194.4
15	24576	3051.7	195.3
16	26112	3063.7	196.1
17	27648	3074.3	196.8
18	29184	3083.8	197.4
19	30720	3092.4	197.9
20	32256	3100.1	198.4
21	33792	3107.2	198.9
22	35328	3113.6	199.3
23	36864	3119.5	199.7
24	38400	3125	200
25	39936	3130.0	200.3
26	41472	3134.6	200.6
27	43008	3138.9	200.9
28	44544	3142.9	201.1
29	46080	3146.7	201.4
30	47616	3150.2	201.6
31	49152	3153.4	201.8

Each CK value is assumed to be 8 bytes long as this is the key length required by the block ciphers if DES or the proprietary CSA algorithm is used. The maximum number of CK values that can be delivered by one APDU is $\lfloor 255/8 \rfloor = 31$ (i.e. the maximum data size possible in one APDU divided by 8). If this algorithm is replaced with an AES the same table can be used. In this case the number of CKs per APDU will be divided by 2, so only the rows corresponding to an even number of CKs per APDU need be considered. This is because an AES key will require twice the number of bytes as a DES key.

These calculations are based on a smart card with an external clock of 5 MHz and an ETU* of 16 clock cycles, which is the fastest speed provided for in the ISO standards [23]. A faster clock speed can sometimes be used but the behaviour of a smart card cannot be predicted (some smart cards will refuse to function), this case is therefore not taken into account.

The values given in Table 1 also assume that data is constantly being sent or received across the I/O between the CAM and the smart card i.e. processing time is not taken into account. This requires that the I/O is being conducted by an UART** in the smart card and the block cipher is being done with a crypto-coprocessor i.e. the I/O and algorithm calculation are not dependent on the CPU. The CPU is therefore just required to send data from the UART to the crypto-coprocessor and *vice versa*, as the UART and crypto-coprocessor will be separate blocks on the chip. In the case of proprietary algorithms a hardware implementation is unlikely to be available, which will significantly lower the amount of keys that can be delivered per second, as CPU cycles will need to be used to calculate the block cipher. It will still be possible to calculate values of CK while the UART is communicating but the performance will be significantly slower than a hardware implementation. As the performance of the proprietary CSA algorithm on a smart card is not known it is not possible to predict the effect this will have on the proposed countermeasure.

To use a smart card in this manner will require a special mode, where the smart card will only respond to commands asking for more keys or to exit this mode. Otherwise the command dispatcher will take too much time and the performance will drop.

As can be seen the bandwidth required to share the CKs is greatly increased from 40 bits per second. It may still be feasible for an attacker to share the series of CKs with one other person if they have a fast enough internet connection, but will be unable to act as a server for numerous people. The client OSTB will also be obliged to decipher the same broadcast as the server OSTB. Even if an attacker is sharing this data with one other person there are likely to be synchronisation difficulties streaming this data from one receiver to another,

* An ETU is an Elementary Time Unit in the $T = 0$ protocol [23] and is the amount of clock cycles required to send one byte. 12 ETU's are required to send 1 byte.

** A Universal Asynchronous Receiver-Transmitter (UART) is an autonomous block on the chip that will receive and and send signals on the I/O pin based on instructions from the CPU. This greatly simplifies I/O routines as the CPU does not need to concern itself with the state of the I/O pin at any given time.

which will lower the quality of the signal that can be produced by the client receiver.

3.2 Using Fast Protocols

The countermeasure proposed above is based on a standard smart card using the $T = 0$ protocol. The smart card industry is currently working on several solutions to the bandwidth problems posed by this protocol. With a fast protocol it should be possible to have a card that deciphers the broadcast on-the-fly. These are summarised below, with a brief description on how this would change or replace the proposed countermeasure:

Proprietary Protocols: A method of using a standard smart card with a proprietary protocol, and therefore a proprietary reader, was presented by Gemplus at Cartes 2003 [12]. This protocol allowed music from a CD player to be deciphered on-the-fly. Using this technology to implement the above countermeasure a smart card would be able to produce 11000 keys per second [12], given the bandwidth required to decipher a CD on-the-fly, which gives a required bandwidth to share the series of CKs of 680.8 Kbits per second. This is potentially too fast as the CK will change 3 times for each DVB packet treated by the CAM, assuming that the broadcast is arriving at its maximum bandwidth of 4 Mbits/s. It may not be possible to re-key the CAM this often. The major draw back is the use of a proprietary protocol means that substantial changes will be required in the CAM to be able to communicate with this card.

USB Smart Cards: Some chip manufacturers propose smart cards with a USB interface that will allow for a larger bandwidth between a smart card and reader [28]. These chips include a USB interface conforming to USB 1.1 that will provide a bandwidth of 10 Mbits/s. This is more than enough to handle a DVB broadcast, although problems may arise with deciphering the signal. The specification of an example smart card chip can be found at [27] based on ST Microelectronic's ST19 chip family. The internal clock frequency can be raised to 10 MHz, which will not be enough to decipher a broadcast at 4 Mbits/s on-the-fly. If we assume that a hardware DES takes 16 clock cycles (i.e. 1 clock cycle per round) the deciphering will take about a tenth of the CPU time, leaving enough time for data transfer etc. In the case of proprietary protocols this chip is unlikely to be able to provide deciphering on-the-fly due to the amount of CPU time that will be required. In practice industry has found these cards inadequate to increase the bandwidth between a smart card and reader due to the complexity of the host interface [25], although this view may no longer be valid.

Secure MultiMediaCards: A more recent initiative is the Secure MultiMediaCards (Secure-MMC) [10] that aims to blend smart card technology with MultiMediaCards [3]. These chips aim to provide secure storage in devices, such as mobile phones, principally for digital rights management. The advantage of this technology in the context of this paper is that MultiMediaCards

generally have a bus rather than a serial interface. The MMC standard allows for a bandwidth of up to 416 Mbits/s depending on the clock frequency and the size of the bus used. It is assumed that a Secure-MMC will be able to decipher several megabytes per second on-the-fly [24]. The Secure-MMC is a relatively new technology so no specifications are currently available. It is assumed that such cards will be able to decipher broadcasts on-the-fly as the new generation of MultiMediaCards are designed to accept clock speeds up to 52 MHz [4].

The use of any of these fast protocols is going to be a difficult choice for a broadcaster. The use of feature rich chips increases the price of each smart card, as each extra block will require more silicon and development time. This extra cost will have to be included in the subscription fees, which may drive customers away. However, it is anticipated that revenue would increase over time if such a solution was chosen as only subscribers would be able to view broadcasts. The proposed countermeasure will provide the most cost effective solution until USB cards or Secure-MMCs become more affordable.

4 Connection Speeds

There are several different connection speeds offered by internet providers. A summary of these connections is shown in Table 2, where the majority of the information is taken from [14].

Table 2. The bandwidth available with different connection types.

Type of Internet Connection	Download Bandwidth Kbits/s	Upload Bandwidth Kbits/s
ADSL 256	256	128
ADSL 512	512	128
ADSL 1024	1024	256
T1	1500	1500
T3	45000	45000

The proposed countermeasure should be effective in stopping the card sharing attack for ADSL users. A smart card with over 2 keys per APDU will easily be a able to saturate the upload bandwidth of a "slow" speed ADSL connection. An ADSL connection with an upload bandwidth of 256 Kbits/s is more problematic. In theory this would make it possible to share one channel with one other person. However, in practice it is unlikely to to be practical as ADSL internet connections will not consistently attain their theoretical maximums. The headers and footers of all the protocol layers will also add to the bandwidth requirements.

This countermeasure will not stop a the card sharing attack where an attacker has access to a T1/T3 connection. These connections provide enough bandwidth

that the stream of keys could be shared with another user. It is assumed that each extra client will add the same bandwidth requirements as the same data needs to be sent to each client. A T3 connection would therefore be able to supply key information to a small group of clients. The proposed countermeasure will not prevent the card sharing attack in this case.

This does not mean that the countermeasure is worthless, as T1/T3 connections are generally only used by businesses. There is also no way of preventing a user with a T3 connection from sharing the broadcast they are watching with at least one other user. Broadcasts are delivered to a user with a bandwidth of between 1 and 4 Mbits/s. An attacker could potentially decipher the broadcast and deliver it in clear to a third party.

5 Conclusion

A method of inhibiting the card sharing attack is described that functions by increasing the bandwidth required to the point where it is less practical to share the information required to conduct the attack. The communication with the card is saturated so the only information that an attacker is able to share is the broadcast being watched rather than an arbitrary channel. It has been shown that sharing the information required to continue conducting the attack is prohibitive unless the attacker uses a T1/T3 connection, which are normally only used by businesses and are not affordable by everybody.

The proposed countermeasure provides a way of inhibiting the card sharing attack until USB and Secure-MMC devices become readily available and affordable. For this reason the countermeasure has been designed to minimise the impact on the existing protocol as major changes to the protocol will be expensive, and may be unnecessary if more powerful secure devices are going to be used in the near future.

The principle problem of using this countermeasure is that one smart card is required per screen. It will not be possible to view one broadcast and video another, or have two televisions viewing different broadcasts, etc. In order to record a second broadcast a viewer would be required to store the data and CWs and have this deciphered on-the-fly at viewing time. This is a possible advantage for broadcasters as they are sure that only legitimate users can view their emissions, as a smart card needs to be present.

References

1. ETR 154:. Digital video broadcasting (DVB): Implementation guidelines for the use of MPEG-2 systems; video and audio in satellite, cable and terrestrial broadcasting applications.
2. Anonymous. CSA – known facts and speculations. `http://CSA.irde.to`.
3. MultiMediaCard Association. `http://www.mmca.org`.
4. MultiMediaCard Association. Application note, an0501-1.00, April 2005. `http://www.mmca.org/compliance/buy_spec/AN_MMCA050419.pdf`.

5. EBU Project Group B/CA. Functional model of a conditional access system. EBU technical Review, Winter 1995.
6. CENELEC. Common interface specification for conditional access and other digital video broadcasting decoder applications. Technical Report CENELEC Standard 50221, European Committee for Electrotechnical Standardization (CENELEC), Brussels, Belgium, February 1997.
7. V. Chachiere. Man ordered to pay $180m restitution for TV signal piracy. Naples Daily News. http://www.naplesnews.com.
8. Cryptoworks. http://www.digitalnetworks.philips.com.
9. D. J. Cutts. DVB conditional access. *IEE Electronics and Communications Engineering Journal*, 9(1):21–27, February 1997.
10. Giesecke & Devrient. Secure and mobile storage media – the memory card with smart card technology. http://www.gi-de.com/, 2005.
11. ETSI. Digital video broadcasting (DVB); support for use of scrambling and conditional access (CA) within digital broadcasting systems. Technical Report ETSI Technical Report ETR 289, European Telecommunications Standards Institute (ETSI), Sophia Antipolis, France, October 1996.
12. Gemplus. Cryptomotion. presented at Cartes 2003, 2003. review available at http://www.prnewswire.co.uk/cgi/news/release?id=112260.
13. L. C. Guillou. Smart cards and conditional access. In T. Beth, N. Cot, and I. Ingemarsson, editors, *Advances in Cryptology - EUROCRYPT '84*, volume 209 of *Lecture Notes in Computer Science*, pages 480–485. Springer-Verlag, 1984.
14. helpwithpcs.com. Internet connections explained, a guide to dial-up, adsl and cable connections. http://www.helpwithpcs.com/internet/internet-connections.htm\#adsl-conn%ections.
15. R. Hewitt. North american MPEG-2 information, July 2003. http://www.coolstf.com/mpeg.
16. D. Holankar and M. Stamp. Secure streaming media and digital rights management. In *Proceedings of the 2004 Hawaii International Conference on Computer Science*, pages 85–96. ACM Press, 2004.
17. Irdeto. http://www.irdetoaccess.com.
18. P. Kalina. No-pay TV costs industry $50m. The Age Journal. http://www.theage.com.um.
19. D. W. Kravitz and D. M. Goldschlag. Conditional access concepts and principles. In M. K. Franklin, editor, *Financial Cryptography – FC '99*, volume 1648 of *Lecture Notes in Computer Science*, pages 158–172. Springer-Verlag, 1999.
20. M. Kuhn. Attack on pay-tv access control systems. Security Seminar talk. University of Cambridge, London, UK., 1997.
21. G. C. Langelaar. Overview of protection methods in existing TV and storage devices. Technical University of Delft, July 1996.
22. J. McCormac. European scrambling system. Waterford University Press, 1996.
23. International Standards Organisation. ISO7816–3 smart card standard: Part 3: Electronic signals and transmission protocols.
24. D. Praca. Next generation smart card: New features, new architecture and system integration. 6th e-Smart Conference, Sophia Antipolis, France, September 2005.
25. D. Praca and C. Barral. From smart cards to smart objects: The road to new smart card technologies. *Computer networks*, 36(4):381–389, July 2001.
26. Seca. http://www.securityit.com.
27. STmicroelectronics. Smartcard solutions ST19 multi-application smartcard ICs. http://www.st.com.

28. STmicroelectronics. STmicroelectronics delivers world's first USB-certified smart card chips. http://www.st.com, 2002.
29. VIAccess. http://www.viaccess.com.
30. Dream Multimedia Worldwide. Dreambox DM7000s user manual. http://www.dream-multimedia-tv.de/manual/manual_eng.zip.