Why AACS will fail

Frank A. Stevenson, January 2007

fstevenson@gmail.com

AACS, Advanced access control system, is the heir to CSS. Content Scrambling System was supposed to protect DVD movies. Although AACS is cryptographically much stronger, it may still fail as an effective countermeasure to piracy on economically grounds. AACS is examined with a view on the economics of attack and defence, and it is shown that the defender is at a significant disadvantage, which may in turn lead to the system being effectively "broken".

Some years ago the DVD format was introduced, and with it the first DRM system for pre-recorded media. The DRM system CSS, was quickly broken [1]. And even though CSS did have a key revocation scheme, only 1 key is known to ever have been revoked. That was the Xing key, which was used in DeCSS, the very first player key that was compromised and made public. But this was largely only a token measure as the system itself was so compromised that in effect all player keys could easily have been calculated, if one could only be bothered.

The copyright industry now seem to have learned some of their lesson, and have abandoned home brew cryptography. CSS' successor AACS is built with industrial strength cryptographic primitives. AES features prominently, and the system is unlikely to suffer a catastrophic cryptographic collapse like CSS did.

But it is still an open question how much protection AACS will provide to the copyright industry. This paper will promote the arguments holding that the protection offered will be largely ineffective. These arguments will be derived from two sources. One the threat model, and secondly an economical comparison of the cost associated with revoking player keys in relation with the cost of finding or 'cracking' new player keys.

Current status

A contributor to the doom9 forums named Muslix64, recently released proof-of-concept code that a AACS protected movie can be decrypted in software if one has access to a title key. It is not known whether Muslix64 has obtained access to player key. Unauthorized playback is now possible with the correct title key, and it can be expected that the same will be possible with a player key within a short time frame.

A short notice on terminology is in place here. Technically the player key is used to determine a volume key, which in turn is used to determine the title keys. For the purpose of this study title and volume keys are largely interchangeable, but only title key will be used, although it might refer to a volume key, which is used to determine the title key.

Player key granularity

AACS uses subset difference trees [2] to simplify player key management. In theory there can be 2^{32}, i.e.4 billion different player keys. The player keys are used to calculate a volume key, which is used to determine a HD DVDs title key(s). The title key is then in turn used to decrypt the copyrighted content of the recorded media. A HD DVD has a capacity of around 25 Gigabyte, which isn't even enough to store 4 billion encrypted 128 bit keys. However the real limiting factor appears to found in the MKB, Media Key Block. The record holding the encrypted title keys, has a maximum size of 16 megabytes, which yields a total maximum of 1 million player keys. These keys are a set covering the valid (non-revoked) player keys.

If someone manages to extract the full set of player keys from a device and makes them public, it can be assumed that the copyright industry will try to revoke such keys in order to prevent newly released titles from instantly appearing in full HD resolution on the net. Using set covering techniques such as described in [5] it is possible to to revoke around 10.000 sets of player keys in total before collateral damage arises, as the number of player devices is beyond the granularity afforded by the MKB block. Moreover, assuming that these disabled devices are under warranty, the consumer that finds his new player unable to play the latest Hollywood blockbuster movie has a strong case for getting his player repaired, or replaced. Such compensation from collateral damage will cost the manufacturers real money. Direct cost is incurred through replacement and repair, and indirect cost are born through damaged reputation with respect to reliability.

Dual and triple roles

The roster of AACS members contain major names from the entertainment and electronics industry. And one can assume they all have some form of nexus with the AACS licensing authority which gives the orders to revoke particular keys. Sony in particular has an interesting role as both a copyright owner, and a device manufacturer, and a key question is how big a cost they are willing to stomach from replacing and fixing devices in order to keep some semblance of security of their copyrighted works.

The cost of obtaining fresh keys

The cost of breaching a security system is a measurable quantity, in terms of money, time and assets, both instrumental and intellectual. The spectre of who can afford to penetrate such a system runs the gamut of casual users, dedicated individuals, well funded individuals, corporates and government institutions. Historically systems of similar security requirements, such as European pay TV have been breached by dedicated individuals [3] But even though the initial cost of obtaining a fresh key may be high, replicating the attack on an identical device with a different set of keys will be lower, and the marginal cost can in many cases approach that of the unit price for the device. This puts the defender at an economical disadvantage if the cost of revoking a key is higher than obtaining one.

Device class attacks

History of Microsoft's gaming console Xbox holds a relevant lesson in relation to the cost of attack. The Xbox as shipped from Microsoft would only run signed executables from original media. But there was much interest in running copied games, or making the device run Linux or other "non-approved" applications. After some time a way of doing this was made public on the Internet. The save feature of a game had a buffer over-run flaw, and by correctly modifying a save game on a memory cartridge, it was possible to boot the machine with Linux and flash the BIOS so the Xbox would no longer require applications to be correctly signed.

The MKB construct allows a total budget of around 10.000 player key set revocations before the revocation scheme melts down. The use of duplicate key sets for a class (make or model) of a player is of significance when addressing the crucial question of how many device class attacks AACS can withstand. The options offered by AACS is a real catch 22. Duplicate key sets among devices, offers no reduction of cost in the case of a device class attack, as the entire class would have to disabled, and only serves to increase the cost of revoking a set of keys if a single device has been compromised. If there are close to 10.000 unique keys within a class, revocation of a whole class becomes nearly impossible, as it will spend the entire revocation budget, and no further revocations are possible. [4] indicates that AACS is not equipped to handle revocation of a device class at all, but instead relies on SPDC, a self professed security through obscurity reptilian oil.

The economical significance of this type of attack which effects a whole class of devices is that that a great number of affected devices are in the market, and the marginal cost of compromising an additional device is very low. If one were to translate this scenario to the world of Blue-ray / HD DVD players, imagine a player that has a picture viewing function. If such a device has a buffer overrun bug, a correctly malformed jpeg image loaded onto a memory card, could when viewed dump all device keys to screen or the card. The marginal cost of obtaining a player key in this case as almost zero, while the total cost of revoking and replacing all affected devices easily will run into the millions of dollars. Granted that device class revocation is at all possible, one could imagine that the AACS licensing authority would be willing to sacrifice a small pacific rim licensee, but would Sony incur this cost on themselves ? This scenario would possibly be give a clue as to what the real monetary cost of "piracy" is. It has long been suspected that they are significantly less than the billion dollar figures which are thrown around with abandon.

The backorder problem

A compromised key will not stop working, even when revoked. It will only be useless for decrypting future titles. Thus the value of a revoked key is a non-zero entity, and can be assigned a value relating to how many titles it will actually decrypt. A recently revoked key is worth more that a key that has been revoked for a much longer time.

The title key database problem

During the process of decoding a title, the player key is only needed for the initial decryption. After that all the work is done using a title key. It can be reasonably assumed that this key will be even easier to recover since it must always be in memory in some form. By building a database that maps disk titles to their respective title keys, a "benevolent hacker" can allow others unlicensed playback of disks, with little risk of disclosing which player key has been compromised. Such a database would be relatively small, and easily distributable on the Internet. Muslix64 could if he wanted contribute to such a title key database, but the legality of this quite dubious, especially under DMCAesque legislation.

Type of attackers

- **Long John Silver**, trades HD movies in closed networks. In order to download movies that he doesn't already have, he must upload new material to maintain his quota. If John obtains a player key, he will guard it well, making sure that the it doesn't get revoked, cutting of his supply of Bollywood blockbusters. The movies that John decrypt will eventually filter out into more open channels. John can obtain his player key from a device that he has hacked himself, or through a device class compromise. It is also possible that his circle of friends can supply with a key, in cases where John doesn't by himself have the resources to produce a key. Unless the AACS regime can identify the compromised key, they are pretty defenceless against Long John Silver and his likes.

- **Linux Hack**, is a consumer whose main motivation for breaching AACS security is mainly a desire to play HD movies as legally as possible on her Linux machine. Apart from "piracy, the better choice" option, waiting for a licensed player is not an option. Judging from experience, an authorized DVD movie player took years to become available for Linux, by which time dozens of unofficial players had been around ever since the wait began. The Linux Hack will probably be willing to go to considerable lengths in order to obtain a working player key. Device class attacks that involves physical interventions into a licensed player should very well be expected to be within the reach of a Linux user. The back order problem, will make even a revoked key attractive. Access to a title key database would also be acceptable for Linux playback. AACS can not completely deny a Linux user the ability to playback protected content.

- **Casual User**, is a consumer who desires to make legal backups of titles that may easily be damaged such as children's movies, or produce unauthorized copies in low volumes. Revoked keys, or title key databases may be available in conjunction with duplication tools. But access to keys that will decrypt the very latest releases will expectedly by restricted as long as AACS can afford to keep up the game of revoking keys.

- **Manufacturer of unlicensed player**, may want to manufacture otherwise compliant devices, but avoid the licensing costs to the AACS consortium. He will do so by duplicating player keys from a licensed player. The problem is that if AACS gets a hold of such a manufacturer they can both sue for patent infringement, and revoke the keys. The AACS have pretty strong defences against hardware manufacturers that will try to avoid the licensing costs.

- **Notoriety seeker** is a highly skilled computer user, whose motivation is to seek a name for himself. This can often be achieved by disclosing devastating security flaws. A device class compromise would fit the bill for such a bombshell. It can reasonable be expected that player class compromises will be published on the Internet, even if it carries no pecuniary reward for the discoverer, but rather the threat of serious litigation. The threat of litigation will only be a modest deterrent for such disclosure. The discovery could for instance be published through anonymous channels, but still be tied to a PGP derived identity. Notoriety and anonymity are not mutually exclusive, even in a day and age of rampant surveillance on the Internet.

Countermeasures

Much of this is not news [4]. But it is my position that the economical disparity between attacker and defender has been underestimated. Even though ISS hints at this, the recommendations that cryptographic functions be moved into protected ASIC circuitry, does not appear to have been followed, as general purpose computer playback is still allowed. Even if the ASIC path is taken, and keys are stored in write only memory, almost anything less than a player build around a single tamper proofed chip may prove inadequate.

Conclusion

Far from being the panacea to protecting digital content, AACS protection will be a shambolic affair, The cracks in the fortress are just beginning to show. My prediction is that as a minimum: Through the major part of the life cycle, unlicensed playback of the majority of titles will be possible. Neither will AACS be able to prevent newly released titles from appearing on the net, including closed forums, within weeks of release.

There is also a considerable risk that expenses associated with key revocation will be so high that further revocations are not a feasible option, in which case the system should be considered broken, not on technical grounds, but rather for economical reasons. A more spectacular scenario would be if the Internet community collected more than 10.000 individual sets of player keys, providing enough coverage of the key set, to effectively render AACS truly broken.

[1] DeCSS
[2] AACS: Introduction and Common Cryptographic Elements
http://www.aacsla.com/specifications/specs091/AACS_Spec_Common_0.91.pdf
[3] Kuhn
http://www.cl.cam.ac.uk/~mgk25/publications.html
[4] Content protection for optical media
http://securityevaluators.com/eval/spdc_aacs_2005.pdf
[5] Revocation and Tracing Schemes for Stateless Receivers (Dalit Naor, Moni Naor, Jeff Lottspiech, 2004)
http://citeseer.ist.psu.edu/rd/20178014%2C502910%2C1%2C0.25%2CDownload/http%3AqSqqSqwww.wisdom.weizmann.ac.ilqSqpeopleqSqhomepagesqSqnaorqSqPAPERSqSq2nl.pdf