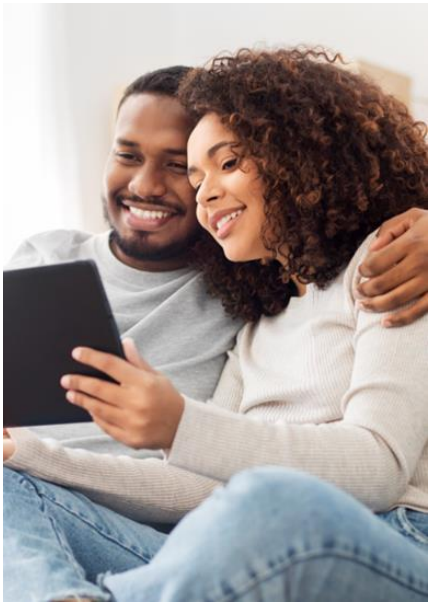
Cartesian®

Video Streaming Services:

Is Content Leakage an Inevitable Fate?

March 2024

1. Introduction



In 2018, a pivotal shift occurred in media consumption as the global number of video streaming subscribers surpassed that of pay TV subscribers. However, this growth has been overshadowed by the rise of digital piracy, posing a significant threat to the business of legal streaming services. According to Dataprot.net, the annual global revenue loss in the movie industry due to digital piracy is estimated between $40 and $97 billion.

Moreover, online piracy of audiovisual content continues to escalate year over year, facilitated by the internet, where users can increasingly find high-quality pirated content on professional-looking illegal services.

But is piracy of video streaming services an inevitable fate that no technical solution can prevent? Or is piracy the consequence of weaknesses in implementations that could be avoided with better thought-out designs and moderate extra effort?

Cartesian, formerly Farncombe, have carried out countless Farncombe Security™ Audits on content protection solutions (CAS or DRM), multi-DRM services, streaming solutions, and end-to-end content distribution platforms. This has allowed Cartesian's Security Team to acquire a realistic and precise understanding of the security level of existing systems, and their vulnerabilities.

By delivering concrete and actionable recommendations to address each identified weakness, and by guiding our customers to secure their systems, Cartesian has a comprehensive understanding of how to address these security concerns, and of the complexity and extent of effort required.

In this paper, we will review the main types of attacks that are used by pirates to build their illegal services, and the vulnerabilities that make them possible. We will demonstrate that realistic and affordable solutions exist to effectively eliminate these weaknesses, or to render them significantly more challenging to exploit, and therefore thwarting illegal services from providing high-quality service. Amongst others, we will explain how to make it impractical for attackers to steal, forge or share access tokens, perform CDN leeching, gain content access by spoofing legitimate users or devices, get well-formed DRM licenses due to forged or stolen tokens, leverage keys leaked from weak devices, or circumvent geo-restrictions.

2. Can Content Leakage Be Prevented?

When considering the security of content streaming services, a widespread view is that preventing content leakage is impossible, no matter the amount of effort made to secure content distribution platforms. This conviction is broadly shared, and deeply ingrained in the collective unconscious.

Regardless of whether it is right or wrong, this popular belief carries significant negative consequences, often serving as a justification for not investing sufficient effort to enhance the resilience of content streaming platforms against potential attacks.

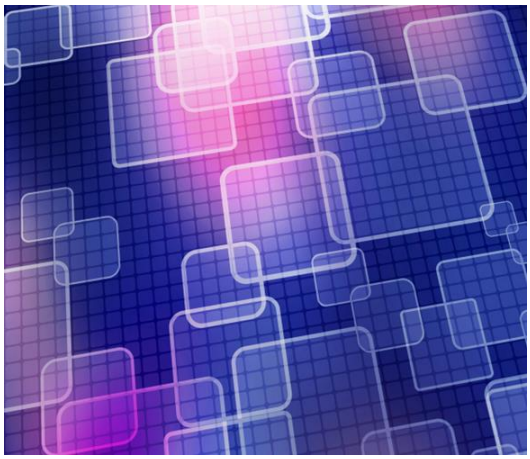
Strictly speaking, this belief cannot be completely disproven. Firstly, it is widely acknowledged that absolute security of IT infrastructures is an unattainable ideal, since all IT infrastructures are built on a plethora of complex components in which vulnerabilities are constantly discovered. Given that content consumption has essentially moved online, content distribution platforms can be subject to the same types of cyberattacks as any other online services.

Furthermore, certain vulnerabilities are difficult to eliminate in streaming services: for example, as will be detailed in section 9, there is no inviolable solution to securing HDMI outputs, since even the most recent versions of HDCP (High-bandwidth Digital Content Protection) can be circumvented.

But intrusions in IT infrastructures and HDMI captures may not be the methods most employed by pirates, who need efficient, easily automated, and scalable attacks. These are typically made possible by vulnerabilities in the legal streaming services.

As a result, we firmly believe that there is significant room for improvement in video service platforms, making it more challenging for pirates to steal content. Some types of attacks can be effectively prevented while others can be made significantly more challenging to execute, less scalable, and less reliable. This would result in reduced service reliability, lower quality, and a limited range of content available for pirate services, making them much less attractive. In the following sections, we will detail several solutions to combat these issues.

3. Content Key Extraction



Most paid-for online streaming services protect their content using the same method: content streams are encrypted before distribution, and the content encryption keys are securely delivered to each authorized device by using one or more Digital Rights Management (DRM) solutions.

The principle of DRM solutions is to embed the ciphered content encryption keys in secured data objects called “DRM licenses”. When a user requests to start a playback, the user’s authorizations are verified and the license server generates a DRM license, specifically targeted to the user’s device, and from which only this device can retrieve the content encryption keys.

Ideally, deciphering the content encryption keys, passing them to the content descrambler, and descrambling the content will occur in a secure environment in the device’s chipset, so that it is impractical for an attacker to access the decrypted keys or the decrypted content. This is generally the case when the DRM solution is natively implemented within the device. But some devices do not offer hardware-based security. In cases where software DRM implementations must be used, the security level is substantially lower.

It is understandable that video service providers want to maximize the reach of their services, to make their offer easy to access on as many different devices as possible. This often results in a heterogenous population of retail client devices with diverse security levels. Clearly the weakest devices will be priority targets for attackers aiming at discovering content encryption keys.

For this reason, it is an absolute necessity to segregate keys by content resolution so that the SD, HD and Ultra HD representations of the same piece of content are not encrypted using the same keys. Devices with limited security or known vulnerabilities will only be granted access to keys for SD and sub-SD representations, whereas only devices satisfying the highest security requirements will have access to the keys for Ultra HD.

Many platforms overlook the critical importance of implementing key segregation mechanisms, needlessly exposing HD and UHD content to potential leaks.

4. Attacks on Multi-DRM License Delivery Services

Multi-DRM license delivery services play a crucial role in the security of a platform. Even with the implementation of top-tier DRM solutions, a poorly designed multi-DRM solution can compromise the security of the entire system. The main attacks on such systems are the sharing and the forging of License Tokens.

External multi-DRM license delivery services are often based on so-called “License Tokens”: when a user wants to start a streaming session, the client application connects to the video service provider’s backend, which checks that the user is authorized to view the requested content, and that all usage rules of the service are satisfied. If these checks are positive, the backend requests a License Token to the multi-DRM service, and this token is forwarded to the requesting application.

This token is then used by the application to request a DRM license from the multi-DRM service. It is supposedly protected against illegal forging by a signature and includes information such as device identification and content identification, as well as all the parameters characterizing the authorizations to be granted by the DRM license. However, a poorly designed multi-DRM service can significantly expand the attack surface of an end-to-end video service platform.

First, if the key used to sign the tokens is leaked (be it by an external attacker or by an insider), a pirate may be able forge valid tokens. If the multi-DRM service accepts these tokens, the pirate will be able to request and receive well-formed licenses. Moreover, if the signature key is static, such an attack may remain easily repeatable for a lengthy period of time while remaining unnoticed. To avoid this, a multi-DRM service should ban the use of static keys and reject any tokens it is not the originator of, even if they are well-formed and validly signed.

Another potential attack is to share copies of a License Token with multiple devices, which can then use these copies to request and be delivered valid DRM licenses. This attack can be effectively mitigated through measures like device-binding and anti-replay mechanisms. If these mechanisms are not in place, such an attack can be extremely easy to perform, providing an ideal way for an illegal service to serve its users with valid DRM licenses.

With a detailed understanding of these mechanisms, Cartesian reviews the security of multi-DRM solutions and can help vendors in perfecting their security.

5. Access Token Theft

For most VOD and Live streaming services, users generally login by entering credentials (user ID and password) before being able to use the application to access content.

Once a user has entered their credentials, the application receives a so-called “Access Token”, which is a small data element, signed by the server, keeping track of this login action. The application will use this token as proof of authentication to grant service access to the user until token expiry.

Such tokens are also named "bearer tokens" because access is granted to the bearer of the token. If illegitimate users are given copies of a token, they will be granted access as if they were the original logged-in user. The magnitude of risk is proportional to the lifetime duration of the token.

For media entertainment applications, it is often preferred for users to remain logged in after their initial sign-in, enabling uninterrupted access for days, weeks, or even months, without having to re-enter credentials.

In implementations where Access Tokens are given such a long lifetime, service platforms are highly vulnerable to token theft/sharing attacks, where a large number of illegitimate users can piggyback, for lengthy periods, on legitimate login sessions.

Such implementations do exist, and they need an urgent improvement: Access Tokens must be given the shortest lifetime possible, and a token refresh mechanism must be put in place. With a properly implemented token refresh mechanism, token sharing is literally impossible.

6. CDN Leeching

To distribute their content, illegal services often take advantage of the Content Delivery Network (CDN) infrastructures used by legal service providers. To do so, they simply redirect their users to URLs exposed by legitimate streaming services, to illegally pull content.

Such a practice is called “CDN leeching”, or “hotlinking”.

Generally, content streamed from a CDN is encrypted. Hence, CDN leeching alone is not sufficient and needs to be supplemented with actions such as content key extraction (see section 3), or License Token forging (see section 4).

With CDN leeching, pirate services can deliver the same streaming quality as legitimate ones and benefit from the same reach and scalability, while the related CDN costs will be invoiced to the legitimate service provider.

CDN leeching imposes a triple penalty on legitimate service providers:

- Revenue loss, as non-subscribers gain access to their content libraries,
- Higher CDN costs to cover the data egress generated by these illegal streams,
- Lower service quality for their own subscribers due to the added, unplanned load on their content distribution infrastructures.

Most CDNs support a token authentication mechanism which, when enabled, denies the delivery of content when requests do not include a valid “CDN authentication token”. Such a token is delivered by the legal service backend to each authorized user starting a viewing session. However, similar to Access Tokens (see section 5), if the token is copied and shared with a large number of users, they will all be able to stream the content from the CDN as long as the token is valid.

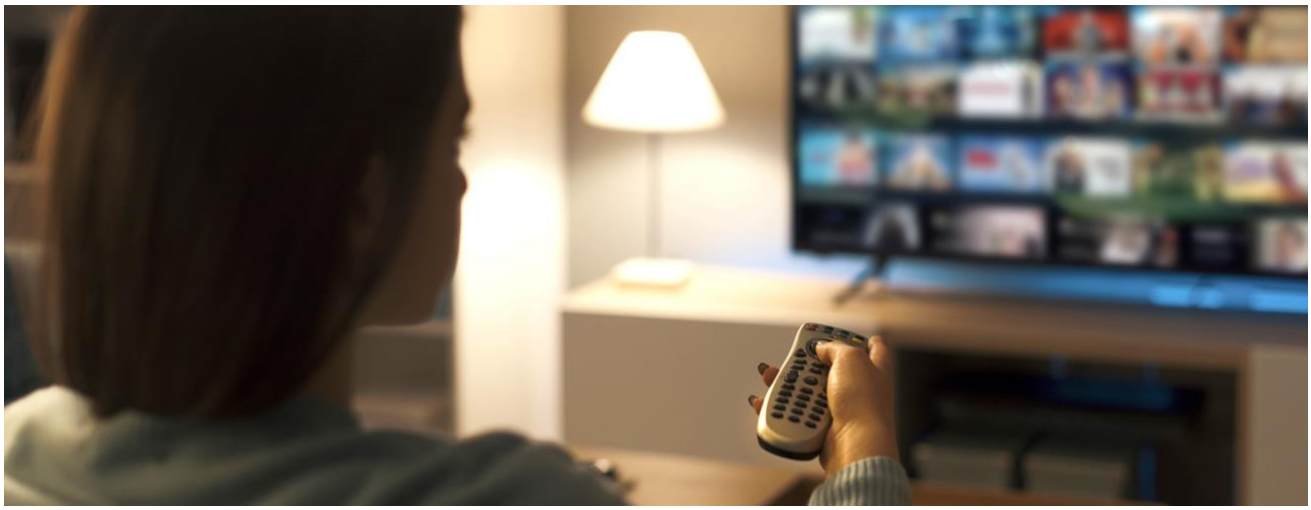
Many service platforms deliver tokens with a lifetime covering the entire viewing session, or even more. This gives plenty of time for these tokens to be widely distributed and extensively used by illegitimate users. This makes CDN leeching extremely easy for pirates, and such implementations need an urgent fix.

The solution is similar to the one described in section 5: like Access Tokens, CDN Tokens must be given a short lifetime (typically a few minutes) so that they need to be refreshed regularly through handshakes with the service backend. With a properly implemented token refresh mechanism, sharing of the CDN Token can be efficiently prevented, making CDN leeching strictly impossible.

7. User-Device Unbinding

Video services provided by satellite operators or Internet Service Providers are generally delivered to so-called “managed devices” (e.g. set-top boxes). Typically, each device is contractually linked to only one subscription contract, hence the operator can easily know which authorizations must be granted or denied to each device.

In online video streaming services, the device/user relationship is more ambiguous: Viewing authorizations are associated with a user account, while the owner of this account can possibly access the service from several different devices, including retail devices (e.g. smartphones), on which the operator has very limited knowledge and control.



This creates a security challenge: technically, it is ultimately the device that must be made capable (or not) of decrypting and presenting a piece of content (through the delivery of a DRM license specifically targeted to that device). Hence, if the service backend does not have a clear and reliable view of the devices used by each user account, it becomes possible for attackers to forge license requests using the authorizations from a legitimate user but authorizing viewing on a device which is in the hands of someone else.

For this reason, there must be a strong binding between each logged in user and the device they use, and this relationship shall be securely traced during the whole lifetime of each login session. When a user requests to play back content, a DRM license shall be delivered to their device only if it is the same device on which they have entered their credentials and are currently logged in.

Cartesian verifies this binding as one of many checks when auditing a DRM solution, a multi-DRM service, or an end-to-end platform. Indeed, ensuring adherence to this principle throughout every aspect of the system is imperative, and it is unfortunately overlooked in many implementations.

8. Client Application Tampering

As previously mentioned, online video streaming services often target a large number of different types of devices, including retail devices that are not managed by the service provider. This commonly includes personal computers (through web browsers), smartphones, tablets, and connected TV sets. This results in relatively unsafe and uncontrolled execution environments. Moreover, these environments may sometimes be even less secure than expected (e.g., rooted or jailbroken smartphones). This is quite different from the case where an operator has full control over the whole device's software stack, including bootloader.

This introduces several additional risks, such as:

- Access to the application code for analysis, which may help pirates in reverse engineering the end-to-end workflows, thus designing smarter attacks,
- Discovery of static keys or secret data buried in the application code,
- Modification of the application code to bypass verifications (e.g., signature or expiry date verifications), masquerade/spoof devices by modifying requests, or add breakpoints to facilitate reverse engineering,
- Co-existence of the application with rogue software, allowing screen capture, data theft, or key leakage.

All these risks must be mitigated by taking the following steps:

- Wherever possible, verifications must be done server-side rather than by the client application, and a "zero-trust" approach shall be applied in the system design.
- The client application code must be obfuscated using specialized, security-oriented obfuscation tools (not just "minifying" tools).
- Code integrity must be checked, including during run-time.
- Trusted Execution Environments (TEE) shall be used wherever possible.
- The native security features of the OS (e.g., Android KeyStore) shall be used wherever possible to minimize the number of cryptographic operations executed in an insecure environment.
- The use of static keys shall be banned, and keys shall, by no means, be buried in the code.
- White Box Cryptography (WBC) shall be used for each cryptographic operation occurring in an insecure environment.
- The application shall verify the environment in which it executes before starting (to avoid, for example, execution on jailbroken devices, or in emulation or debug environments), and this must be complemented by run-time checks.

Following the above principles significantly mitigates the risks that are inherent to addressing unmanaged retail devices and is a key focus in Cartesian's security audits.

9. HDCP Circumvention

As previously mentioned, no inviolable solution exists today to secure HDMI links since even the most recent versions of HDCP can be circumvented.

Hence, each time a video service authorizes viewing on set-top boxes or HDMI dongles, it is technically possible (though illegal) to capture a digital copy of the content from their HDMI output port.

Pirates generally do not use this method for building content-rich illegal VOD services, but commonly use it for leaking content from physical media (e.g. Blu-Rays), as well as for leaking live channels: live video streams are captured in real time from HDMI ports, re-encoded, packaged, and injected in real time in the pirate's own streaming infrastructure.

This method of content leakage is probably the only vulnerability for which no effective defense exists. And even if a newer (hopefully undefeated) version of HDCP is released, it will take time until it is supported by a majority of the deployed retail devices.

Nevertheless, even if not impenetrable, HDCP must be activated in all cases. For streaming services, the DRM solution must be used to enforce a minimum HDCP version, in accordance with the content's resolution.

And to mitigate the relative lack of effectiveness of this protection, forensic video watermarking is strongly recommended. Watermarking is a deterrent against unauthorized distribution and enables the identification of the user account or device responsible for content leakage.

This opens the way to possible legal actions, or, in the case of live, to take down in real time the involved streaming sessions.

However, the selected watermarking technology needs to be robust against attacks, and its end-to-end implementation secure. Farncombe Security™ Robustness Tests and Farncombe Security™ Watermark Implementation Reviews specifically seek to verifying these points.

In particular, collusion attacks (which consist of mixing several watermarked sources, for the purpose of obscuring the watermarked information) are one of the biggest challenges faced by watermarking technologies, and not all solutions offer sufficient resilience.

10. User Account Sharing

The terms and conditions of most online video streaming services specify that access to the service is personal, non-commercial, and shall not be shared outside the user's household.

However, many services are facing user account sharing practices, which are in violation of this clause. With such practices, several households may access the service while the service provider collects only one subscription fee. This fraud can take different forms and be realized on different scales, from small-scale

sharing (e.g., with friends and relatives) to large-scale sharing (e.g., when user account credentials are sold on the internet).

Although small-scale sharing is difficult to totally eradicate (and is sometimes tolerated by the service provider), large-scale sharing constitutes a full-fledged act of piracy and must be combatted by all means.

To do so, a package of good practices must be put in place:

- First, the number of concurrent streaming sessions per user account must be limited. Most service platforms do this, but it is rarely done in an effective manner. Indeed, in many implementations, once a viewing session has started and until the DRM license expires, the backend has no reliable way to know whether the session is still active or has been stopped, and has no reliable means to take it down. Hence, streaming sessions may still be running without the backend even being aware of it. With such poor tracking of streaming sessions, it becomes easy to circumvent concurrent streaming limitations. Such a weakness can be easily addressed thanks to the solutions prescribed in section 6 (to prevent CDN leeching), by imposing regular handshakes between the application and the backend for token refresh.
- In addition, the number of simultaneous login sessions per user account must be limited. When the limit is exceeded, the oldest login session must be taken down (which can be efficiently done only if our prescriptions of section 5 have been followed).
- Lastly, the number of devices used with each user account must be limited, as well as the pace at which new devices are introduced to replace previous ones.

All the above restrictions should be implemented, as they will drastically limit the magnitude of the account sharing issue and efficiently prevent large-scale sharing. However, it might be difficult to find threshold values which totally eradicate abusive usages without degrading the user experience of fair usages within households.

Where a satisfactory trade-off cannot be found, turning to analytics-based solutions becomes essential. These solutions efficiently detect abuses, and identify the accounts involved. Cartesian can help in implementing such a solution (see www.cartesian.com/credential-sharing-mitigation-strategies).

11. Geo-blocking Circumvention

Most online video streaming services implement a geo-blocking mechanism so that the content they distribute can only be accessed from the regions authorized in their licensing agreements.

Such geo-blocking mechanisms are generally based on the IP-address from which the connection originates, which gives an indication of where the user is located. But the reliability of these solutions is uneven, and they can be seriously challenged by VPNs, which are specifically intended, amongst other purposes, to bypass geo-blocking.

The effectiveness of geo-blocking solutions is instrumental to the ecosystem, and deficiencies can have the following consequences:

- Service providers fail to fulfil their content licensing agreements,
- Streaming services may unfairly face competition of similar services from other countries, offered at cheaper prices,
- The exposure of video services to attacks and content leakage is higher.

Hence, it is crucial for service providers to efficiently block access to users who are not located in the eligible region, as well as those hiding their actual location behind a VPN.

To do so, an effective geo-blocking solution must be selected which not only accurately geo-locates users, but also reliably detects the use of VPNs. Such geo-blocking must be performed at two levels:

- At the service backend level, to deny DRM license delivery and display explicit and accurate error messages when the user is not in an eligible location, and,
- At the CDN level, to technically block content access.

Moreover, the geo-location check must not only be performed before allowing the start of a streaming session, but also at regular intervals during streaming.

To enable service providers to confirm the efficiency of the geo-blocking mechanisms they use, Cartesian's Geo-Blocking Circumvention Testing evaluates geo-blocking solutions by testing them against thousands of geographical endpoints, checking the successful identification and blocking of VPNs and other bypass methods.

12. Conclusion

The above analysis is not exhaustive and only reviews the most common attacks used by pirates to leak content and build illegal services. But it shows that, although perfection may not be possible, there is significant potential for improving the security of online streaming platforms and making these attacks either impossible or significantly more difficult to execute.

Why are so many online services still concerned by these easy-to-prevent attacks? Simply put, because prevailing beliefs imply that content security solely relies on the DRM solutions in use, and that content leakage cannot be prevented, regardless of the efforts made. Therefore, while DRM solutions are typically designed and implemented with a strong security focus, the teams behind the end-to-end service platforms primarily prioritize aspects like video quality, user interface, quality of service, and navigation performances, often overlooking the crucial role of the end-to-end workflows in achieving optimal levels of content security.

This paper reveals that ensuring security not only depends on the use of undefeated DRM solutions and on a secure IT infrastructure, it also relies, to a much larger extent, on intricate elements within the end-to-end design such as token lifecycles, authentication and license delivery workflows, user-to-device binding, key management, and enforcement of usage rules. Moreover, all this needs to come in addition to rigorous processes and policies (which are not dealt with in this paper). Hence, assessing the security of such systems cannot rely solely on generic cybersecurity audits or checklist-based assessments. Ticking boxes in a checklist will never be sufficient to ensure that a platform design and the way it is operated are secure.

13. How Cartesian Can Help

Unlike most other audits, Cartesian's Farncombe Security™ Audits are specialized in content distribution and content protection systems, and are based on deep and comprehensive analyses of the audited system's design, its implementation, its operation, and associated processes. They cover all the points addressed above in this paper, and much more.

Farncombe Security™ Audits not only assess the current security levels of content protection solutions, multi-DRM services, streaming solutions, and content distribution platforms, but also provide guidance to service providers and solution vendors on how to achieve resilience against content leakage methods, aligning their systems with the highest security standards.

To complete this range of indispensable Farncombe Security™ services, Cartesian also delivers:

- Geo-blocking circumvention testing,
- Watermark robustness testing,
- Watermark implementation reviews,
- Credential sharing detection, and,
- Specialized content security advisory.

All details on these services can be found on www.cartesian.com/services/content-security.

Cartesian is committed to helping service providers and solution vendors make content safer. To learn more about our services, feel free to contact us at www.cartesian.com/contact.

www.cartesian.com