https://doi.org/10.2352/EI.2026.38.4.MWSF-310

A Practical Approach to Traitor Tracing in Environments with Lossy Feedback Channels

Vladimir D. Živković; Irdeto; Hoofddorp, the Netherlands,
Abhijeet Golwelkar; Irdeto; Denver, Colorado, the USA,
Ronald Peters; Irdeto; Hoofddorp, the Netherlands

Abstract

This paper proposes Uniform Switching Identities (U-SWIDs) as a lightweight, collusion-resistant identity scheme designed for computationally constrained environments. U-SWIDs use uniformly structured identities and simplified penalization logic while retaining effectiveness comparable to classical Tardos-based methods. Through simulations, we show U-SWIDs maintain robustness even under lossy feedback conditions - common in forensic watermarking scenarios. Compared to Approximated Tardos Switching Identities (AT-SWIDs), U-SWIDs offer improved scalability, ease of generation, and operational simplicity without compromising traceability. The findings suggest U-SWIDs are a viable alternative for practical traitor tracing systems, especially where delivery, derivation cost, and resilience to partial symbol loss are critical deployment factors.

Introduction

Video streaming services continue to face persistent threats from piracy, not limited to Subscription Video on Demand (SVOD) or Live Sports, but extending across all forms of valuable video content, including Online Gaming. To address this, the industry increasingly relies on a feedback channel enabled by forensic video watermarking. This mechanism helps trace pirated content back to the original licensed device. However, attackers often counteract by forming collusive coalitions to obscure watermark patterns. While cryptographic research has long explored collusion-resistant methods, this paper contributes practical insights and adaptations to complement and extend this established body of work.

Related Work

Foundational work on traitor tracing began with Fiat and Naor's construction of broadcast encryption schemes, introducing early collusion resistance principles [1]. Boneh and Shaw later formalized fingerprinting codes for digital content protection, strengthening the theoretical basis [2]. Tardos advanced this field with a probabilistic code design achieving optimal length bounds against collusion, which became the gold standard [3]. Subsequent refinements, such as those by Laarhoven, extended Tardos codes for practical implementation [4]. Our work builds on these contributions by proposing Uniform SWIDs—simpler yet effective identifiers suitable for environments where low-complexity and robustness against feedback loss are essential.

Domain Specifics

In both Digital Video Broadcasting (DVB) and Over-The-Top (OTT) video streaming, traitor tracing is commonly implemented through forensic watermarking. As discussed in our previous work [5], when content protection mechanisms are compromised - whether through key leakage or by circumventing the encryption - the sole viable method for identifying the compromised client devices (receivers) is the use of forensic watermarking. Since watermark symbols (or *watermarks*), which encode identity-related bits, are embedded directly into the video content, disabling the encryption does not eliminate these embedded watermarks. *Quod significat*, the identity associated with the misused device remains encoded in the stream and can therefore be traced.

Client vs. Headend-side Forensic Watermarking

Forensic watermarking in content distribution systems is typically implemented using one of two approaches:

1. *Headend-side Forensic Watermarking*, also known as *A/B Watermarking*, and
2. *Client-side Forensic Watermarking*, often referred to as *Overlay-based Watermarking*.

Headend-side (A/B) Watermarking

The *Headend* in *Headend-side (A/B) Watermarking* covers the following stages of a content provider's workflow: **content preparation**, **content protection**, and **content delivery** [5]. Thus, watermarking takes place entirely within the provider's **Secure Perimeter**, rather than on end-user devices (or during **content consumption**). Specifically, *A/B Watermarking* embeds watermark symbols—encoding identity bits—at well-defined content boundaries: media-segment boundaries in OTT streaming or crypto-period boundaries in DVB systems. This produces time-multiplexed identity encoding, or **Switched Identities (SWIDs)**, where the full user identity (or license identifier) is progressively embedded in the video stream. Once encoded, the identity is cyclically repeated to ensure persistent identification. Because watermark symbols must be dynamically switched, A/B Watermarking requires embedding during content preparation, particularly at the bitstream encoding stage, which lies inside the Secure Perimeter. For each segment, frame, or content unit, two watermark variants—*A* and *B*—are pre-generated, allowing binary identity encoding.

Client-side (Overlay-based) Watermarking

Overlay-based Watermarking applies the watermark after bitstream decoding, directly on the client device. The watermark is overlaid onto the video just before rendering to the display, once the content has been demultiplexed, decrypted, and decoded. This method decorrelates the watermark from the encoded bitstream, which has both advantages and disadvantages. On the one hand, a stronger watermark can make it easier to detect during forensic analysis. On the other hand, visible overlays can negatively impact content fidelity, potentially drawing complaints from both end-users and content owners. Moreover, since watermarking occurs outside the Secure Perimeter - after decryption and decoding - this approach introduces a larger attack surface for attempts to remove or circumvent the watermark. A further challenge with overlay-based methods is scalability: unlike *A/B watermarking* which uses two variants per unit of content, overlay-based systems must support N distinct watermark variants, where N corresponds to the number of identifiable devices or licenses.

Collusion vs. Forensic Watermarking

In the context of this work, collusion-attacks present distinct implications for *A/B Watermarking* and *Overlay-based Watermarking*. These impications are most effectively illustrated in Figures 1 and 2.

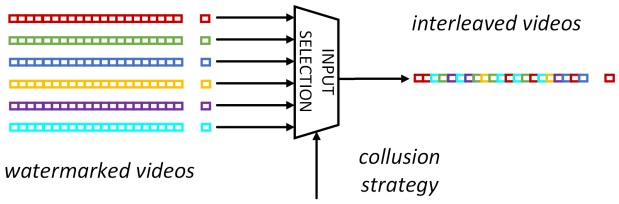


Figure 1. *Interleaving Collusion Attack.*

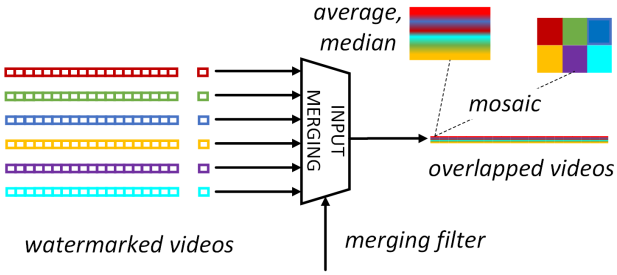


Figure 2. *Overlapping Collusion Attack.*

Video *Interleaving* refers to the mixing of video content segments from the same stream, where each segment has been decoded by a different device (i.e., a different license holder). This form of attack generally has little to no impact on *Overlay-based Watermarking*, but it does affect *A/B Watermarking*. However, even in the presence of interleaving, *A/B Watermarking* is not rendered ineffective; rather, it may require longer content captures to filter out the *accusation noise* introduced by collusion among multiple users.

Video *Overlapping*, on the other hand, involves frame-level mixing, typically at the pixel level. In this scenario, an attacker synchronizes multiple decodings of the same content and combines corresponding pixels - often using operations like median or averaging filters. Because *Overlay-based Watermarking* involves N distinct symbols simultaneously (where N is the number of devices or licenses, hence, N is certainly a large number), the watermark signal is highly susceptible to degradation under such attacks. In contrast, *A/B Watermarking* is far more resilient. Since only two variants (A and B) exist at any given point in time, the majority variant statistically dominates, and the probabilistic construction of **identity codewords** used in traitor tracing makes it extremely difficult for an attacker to fully neutralize the watermark through such averaging methods.

In this paper we focus exclusively on *A/B Watermarking*.

Practical Collusion Cases

When discussing **collusion strategies**, traitor tracing researchers have identified several canonical attack models early on, as outlined in [2]. These include: (1) the *coin-flip* or *random schedule* strategy, (2) *majority voting*, (3) *averaging*, and (4) *min/max selection*. Later work, such as [4], logically extended this set with the (5) *scapegoat* strategy. In the context of video streaming, whether via DVB or OTT platforms, it is reasonable to assume that attackers targeting forensic watermarks cannot reliably decode the **embedded information payload** (i.e., the identity bits), unless there are significant flaws or oversights in the watermarking system. An attacker may detect the **presence** of a watermark but would typically be unable to interpret its encoded identity.

Consequently, several of the strategies from [2] - including *majority vote*, *averaging*, and *min/max* - are functionally equivalent to the **video overlapping attack** illustrated in Figure 2. Implementing attacks such as *majority vote* or *min/max* at the **pixel level**, for each frame (e.g., 50 fps), especially for high-resolution video content (1080p or 4K), would be computationally expensive. Performing such operations in real time requires specialized hardware, and while devices capable of merging multiple synchronized video streams are commercially available, they are costly and typically limited in the number of streams they can handle simultaneously.

In practice, the **most feasible form** of collusion is the use of a **load-balancing** node that distributes decoding across multiple devices. Here, scheduling strategies such as *random selection* or *round-robin* are commonly employed. These approaches correspond to video interleaving attacks, which are particularly relevant for *A/B Watermarking* schemes.

Therefore, in this paper, we narrow the scope of collusion strategies to focus on the *random* and *round-robin* scheduling models, as they represent the most accessible and practical options for attackers.

Model Description

To evaluate our hypothesis, we adopted an empirical approach, building a simulation model and conducting experiments with varying parameters (see Figure 3). We first outline the architecture, detailing its components, inputs, and outputs, followed by a discussion of specific design features.

The model is composed of five primary components:

1. **Headend Node:** The Headend Node models the Secure

Perimeter of the system, where watermarking symbols are embedded during content preparation and protection, switched during content delivery, and subsequently detected from pirate's feedback. It encompasses the essential operations of segment encoding, segment encryption, secure distribution to authorized receivers, and feedback-based watermark detection.

2. **Pirate Node:** Simulates a collusion server applying predefined strategies (*random* or *round-robin*, see Section "Practical Collusion Cases". Functionally, it resembles a load balancer, transparently redistributing content.
3. **Colluder Set (Coalition):** A group of legitimate clients/license holders, often unknowingly[6]. These devices act as black boxes while providing critical feedback for traitor tracing.
4. **Communication Channel:** Connects the components, where **bit loss** is simulated by randomly dropping responses at the headend. Implementation uses IPC via Linux FIFOs to ensure logical isolation.
5. **Accusation Module:** Processes pirate-node feedback, generating either trend reports (evolution of suspicion over time) or final "penalization" reports (shortlists of accused devices).

Experiments' Setup

The experimental setup was designed to evaluate two types of **Switching Identities (SWIDs)**:

1. **Approximated Tardos Identities (AT-SWIDs):** SWIDs generated per the probabilistic model proposed in [4].
2. **Uniform Switching Identities (U-SWIDs):** SWIDs introduced in this paper, constructed as reproducible yet statistically uniform random bit patterns.

Due to implementation constraints, we were unable to fully reproduce the large-scale generation and the corresponding accusation of AT-SWIDs described in [4]. Instead, we use Figure 1(a) and Section 2.6 from [4] as reference benchmarks for comparison. To ensure stricter reliability, our experiments with U-SWIDs adopt error probabilities of 10^{-5} rather than the 10^{-3} used in [4].

For U-SWIDs, a dedicated accusation algorithm was implemented in the **Accusation Module** (see Section "Model Description"). This algorithm computes penalization scores for each user/device, which form the basis for traitor identification. The simulation model establishes identity mappings as follows:

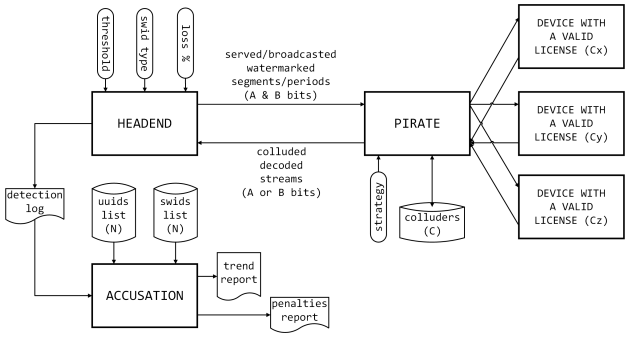


Figure 3. *Collusion Simulation Model.*

- **User/Device Identifiers:** Provided as **Universally Unique Identifiers (UUIDs)** in the standard *8-4-4-4-12* format defined in RFC 4122 [7], each representing a unique device.
- **SWIDs:** Represented as hexadecimal ASCII strings of fixed length (8192 characters, equivalent to 32768 bits), ensuring consistent processing across experiments.
- **Mapping:** Each UUID is associated with a single SWID, enabling one-to-one correspondence. For U-SWIDs, this mapping is immediate; for AT-SWIDs, similar alignment can be adopted in experiments to ensure consistent evaluations.

SWIDs under Computational Constraints

In practice, SWIDs are used by Content Delivery Networks (CDNs) as control patterns to steer switching between *A*- and *B*-variant media segments. CDNs operate under strict constraints, permitting neither computationally intensive processing nor complex execution pipelines, and impose tight memory limits. Consequently, long SWID bitstrings shall be derived on demand rather than stored persistently. AT-SWIDs rely on floating-point–driven functions to enforce U-shaped bit distributions (see later in Figure 4), which are often unsupported in CDN scripting environments. In contrast, U-SWIDs use uniform distributions and can be derived using efficient primitives (e.g., symmetric encryption), making them well suited for practical CDN deployment.

Bit-statistics of different SWID types

As illustrated in Figure 4, the two types of Switching Identities (SWIDs) exhibit distinct bit-level statistical characteristics. These differences arise from the underlying methods used to generate the identities. For the AT-SWIDs, we employ the algorithm described in [4], which results in a **non-uniform distribution** of bits across each position. Specifically, the distribution tends to follow a *U-shaped* pattern, where extreme probabilities—such as those greater than 0.8 or less than 0.2 - are more prevalent, while mid-range probabilities (e.g., $0.45 < p < 0.55$) occur less frequently. This skew is intentional and aligns with the probabilistic design of the Tardos fingerprinting scheme. In contrast, the U-SWIDs are generated using a standard random number **derivation** process, resulting in an **approximately uniform bit distribution** across all positions, with each bit having a near-equal likelihood of being 0 or 1 (i.e., $p \approx 0.5$). It is important to note that the distribution characteristics shown in Figure 4 become even more pronounced when scaling the number of users from 10K over 100K to 1M identities.

U-SWID Construction

U-SWID is designed for straightforward derivation on typical CDNs or end-user device platforms. It is implemented as a 32768-bit AES-128-CBC ciphertext [8], generated by encrypting a plaintext constructed from a specific UUID. This UUID is replicated and concatenated into a 8192-byte ASCII character buffer. The encryption process uses a broadcaster-specific secret key and an associated initialization vector (IV) string. These parameters can vary across different video broadcasters or streaming services. The simplicity of this design offers a practical advantage in lossy feedback environments, where every bit of the SWID holds equal significance - or insignificance. Consequently, partial loss of watermark symbols does not disproportionately impact the effective-

ness of the accusation (penalization) process. This robustness under lossy conditions represents a key contribution of this paper.

U-SWID Penalization

As previously discussed, the process of **penalization** involves updating the scores associated with each UUID in response to feedback received from the pirate node. Later in this paper, we present the corresponding code listing for the U-SWID penalization algorithm analyzed in this study. While the penalization methods may differ depending on the SWID type, they share several common elements:

- `score`: the incremental value that the latest feedback contributes to the cumulative score of a user/device, based on whether the bit values match or differ.
- `detected_variant`: the bit value (0 or 1) extracted from the most recent feedback.
- `user_variant`: the corresponding bit value in the user's SWID at the same position as the `detected_variant`.
- `p0` and `p1`: the probabilities of observing a zero or one at the given bit position.

Future work may include incorporating the AT-SWID penalization algorithm into the model. This will likely require additional parameters once the penalization details are fully specified.

Lossy Feedback

Key performance metrics for forensic watermarking - such as **fidelity** and **robustness** - were identified early on in foundational research (e.g., [9]). However, simultaneously optimizing these metrics often introduces conflicting technical requirements. For instance, content providers may demand imperceptible watermarks to preserve a high level of visual fidelity yet enhancing imperceptibility often compromises the robustness of the watermark against degradation. When additional constraints such as information capacity and implementation complexity are considered, the design space becomes even more constrained. As a result, vendors of watermarking technologies are forced to make trade-offs, frequently leading to the potential loss of watermark symbols.

Such losses may stem from both **intentional attacks** and **non-intentional attacks**, and cannot be overlooked in realistic scenarios. This brings us to the second major contribution of this paper: identifying and modeling the effect of lossy feedback channels on traitor tracing systems. In our simulation model (see Figure 3), this lossiness is accounted for by randomly dropping feedback bits according to a predefined loss factor. Empirical observations suggest that a 0.0% feedback loss rate is unrealistic. In practice, unintentional distortions during the content preparation pipeline (e.g., re-encoding), screen-induced transformations (e.g., downscaling/upscaling), and invasive capture techniques used during forensic investigations all contribute to bit loss. Additional losses may also result from behaviors such as channel surfing, ad breaks, or limited capture durations. Consequently, our simulations account for up to 50.0% loss of watermarking symbols, reflecting these mentioned practical deployment conditions.

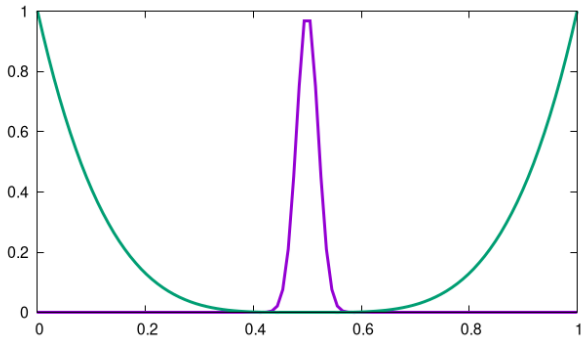


Figure 4. *Distribution characteristics of two different SWID types.*

```
1: /* Uniform Switching Identities Penalization */
2: float uniform_penalization (uint8_t user_variant,
3:                             uint8_t detected_variant,
4:                             float   p0,
5:                             float p1) {
6:     float score = 0.0f;
7:     if ((0 == detected_variant) &&
8:         (user_variant == detected_variant)) {
9:              score = p1/p0; }
10:    else if ((1 == detected_variant) &&
11:             (user_variant == detected_variant)) {
12:             score = p0/p1; }
13:    else if ((0 == detected_variant) &&
14:             (user_variant != detected_variant)) {
15:             score = -p1/p0; }
16:    else { score = -p0/p1; }
17:    return score; } }
```

Results

This paper presents two simulation scenarios using U-SWIDs, focusing on their robustness under feedback loss. For clarity of visualization, the user set size and number of colluders were chosen with practical considerations: (1) 3 colluders out of 10K users with 0% feedback loss, and (2) 3 colluders out of 10K users with 50% feedback loss. In addition, (3) we conduct a scenario representing a large collusion case and compare the obtained U-SWID results with the reported AT-SWID benchmarks from [4].

Visualization Graphs

In the first two scenarios, colluders provide watermarked video segments to a pirate node, which applies a random selection strategy and forwards one response to the headend, where watermark detection is abstracted. Results are presented using two plots: a **trend graph**, showing score evolution across all 10K U-SWIDs, and a **penalization graph**, providing a snapshot of scores at a given time. The threshold curve in the trend graph is updated dynamically per detected bit. The penalization graph highlights outliers by contrasting colluder scores against the broader user population. Together, these visualizations demonstrate the robustness of U-SWIDs under lossy feedback and their practical advantage over AT-SWIDs.

Dynamic Threshold and U-SWIDs

The U-SWID penalization process employs a **dynamic threshold** that adapts to the evolving score distribution. At each feedback step, the threshold is computed as the current average

plus 5.5 standard deviations (selected experimentally, as 3σ produced frequent false positives), enabling reliable separation of typical users from outliers. This threshold retains 99.99866% of user scores below it, consistent with the Six Sigma framework [10], and supports high-confidence colluder identification under lossy feedback.

U-SWIDs with 0.0% Loss

Fig. 5 presents the trend analysis for the first simulation scenario. The threshold curve (violet) remains smooth for U-SWIDs, consistent with the behavior predicted by Tardos [3]. All three colluders are identified after approximately 310 feedback bits, a result that is consistent across simulation runs. The colluder score trajectories remain stable, and the final penalization snapshot (Fig. 6) clearly exposes the three colluders as dominant outliers above the dynamic threshold.

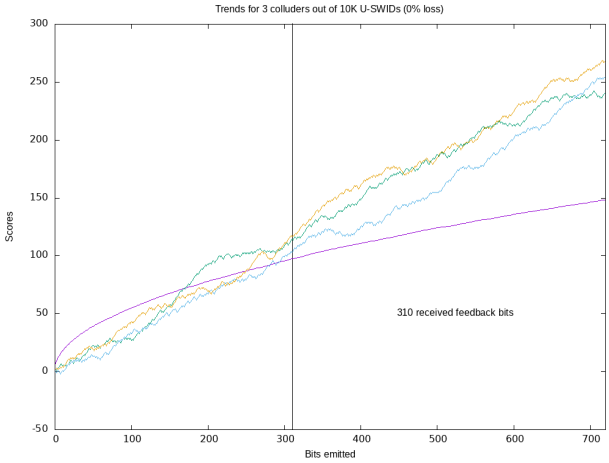


Figure 5. *Trends for the case of 3 colluders with U-SWIDs & 0.0% loss.*

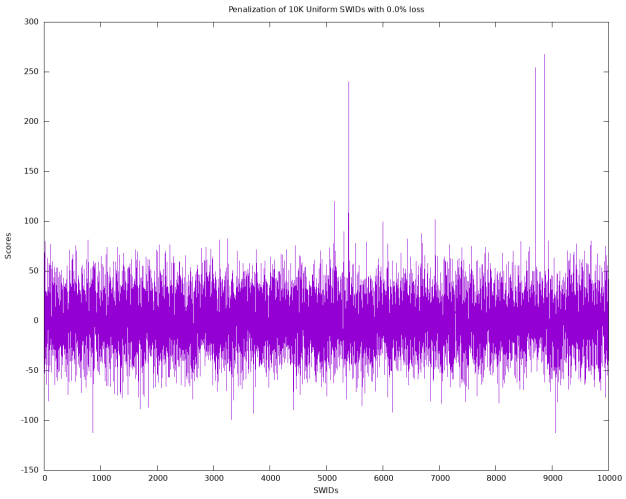


Figure 6. *Penalties for the case of 3 colluders with U-SWIDs & 0.0% loss.*

U-SWIDs with 50.0% Loss

Fig. 7 presents the score evolution for the second simulation scenario, evaluating U-SWIDs under 50% feedback loss. Compared to the lossless case, the threshold curve (violet) becomes less smooth, reflecting increased variance due to missing feedback. Colluder score trajectories exhibit greater fluctuation and slower growth, requiring approximately 700 feedback bits for reliable identification—about 2.3 times more than in the 0% loss case. The penalization snapshot (Fig. 8) shows reduced absolute scores, yet the three colluders remain clearly distinguishable as dominant outliers despite the high loss rate.

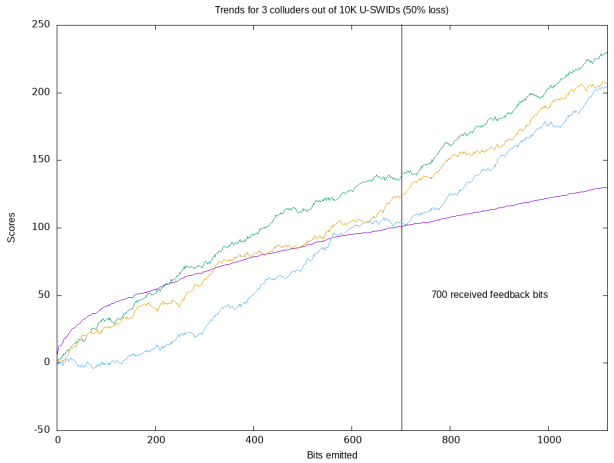


Figure 7. *Trends for the case of 3 colluders with U-SWIDs & 50.0% loss.*

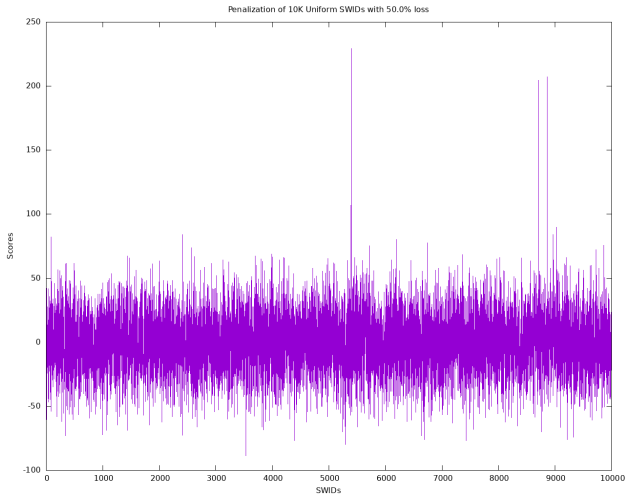


Figure 8. *Penalties for the case of 3 colluders with U-SWIDs & 50.0% loss.*

Large Collusion Case

We compare U-SWIDs with AT-SWIDs [4] in a large-collusion setting, assuming 25 colluders within a user base of 1 million. While [4] reports that interleaving collusion requires 109,585 AT-SWID bits at an error probability of 10^{-3}, our experiments show that U-SWIDs of length 32,768 bits consistently require no more than 31K bits under identical conditions, while operating at a stricter error probability of 10^{-5}.

This constitutes a substantial improvement in practical scenarios (see Section "Practical Collusion Cases"). Polyalphabet

schemes (e.g., A/B/C/D) are impractical due to vulnerability to overlapping collusion (Fig. 2) and excessive computational, communication, and storage costs. The dynamic threshold in Fig. 9, based on a 5.5-sigma criterion, reliably isolates colluders while suppressing noise from regular users; higher thresholds (e.g., 6-sigma) may be applied for larger populations (e.g., 1 billion users) [11].

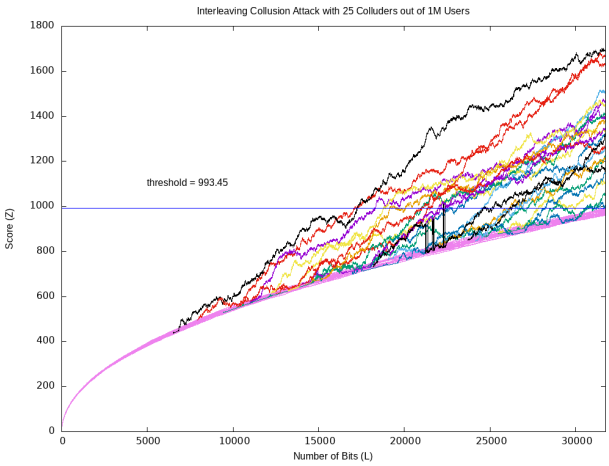


Figure 9. *Large collusion case with U-SWIDs.*

Conclusion

This paper introduced Uniform SWIDs, a lightweight approach for generating collusion-resistant identities in resource-constrained environments. As practical delivery platforms such as CDNs offer limited memory and computational capacity, it is essential that SWIDs can be efficiently derived without degrading the penalization process. U-SWIDs reduce computational overhead while maintaining effectiveness comparable to traditional schemes, with a penalization strategy aligned with Tardos [3] and Laarhoven [4]. Experimental results demonstrate robust performance under both ideal and lossy conditions, particularly against interleaving collusion. Compared to AT-SWIDs, U-SWIDs exhibit improved resilience to feedback loss, underscoring their practical value for real-world traitor tracing in video streaming systems.

Acknowledgments

This work is the culmination of many years dedicated to combating digital content piracy. Special thanks go to the former **Pick'n'Jar** team (2016–2020) for their assistance in implementing and testing ideas related to **Uniform Switching Identity** in practical scenarios.

References

[1] Benny Chor, Amos Fiat, and Moni Naor, Tracing Traitors, Advances in Cryptology - CRYPTO'94, Springer, Berlin, 1994, Pages 257-270, 10.1007/3-540-48658-5_25.

[2] Dan Boneh, and James Shaw, Collusion-secure fingerprinting for digital data, IEEE Transactions on Information Theory, Volume 44, Number 5, 1998, Pages 1897-1905, 10.1109/18.705568.

[3] Gábor Tardos, Optimal probabilistic fingerprint codes, Journal of the ACM (JACM), Volume 55, Number 2, Pages 10, 2008, ACM.

[4] Thijs Laarhoven, Jeroen Doumen, Peter Roelse, Boris Škorić, and Benne de Weger, Dynamic Tardos Traitor Tracing Schemes, IEEE Transactions on Information Theory, Volume 59, Number 7, 2013, Pages 4230-4242, 10.1109/TIT.2013.2251756.

[5] Vladimir D. Živković and Ronald Peters, Sidecar-based VOD Byte-range AB-Watermarking End-to-end Sandbox System with Encoder Integration, MHV '25: Proceedings of the 4th Mile-High Video Conference, 10.1145/3715675.3715797, ACM, New York, NY, 2025.

[6] David Buchanan, 2024, MPEG-CENC: Defective by Specification, https://phrack.org/issues/71/6.html#article.

[7] P. Leach, M. Mealling, and R. Salz, A Universally Unique Identifier (UUID) URN Namespace, Network Working Group, July 2005, https://datatracker.ietf.org/doc/html/rfc4122.

[8] NIST, 2001, The Advanced Encryption Standard (AES) (FIPS PUB 197), US Department of Commerce, https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf

[9] Lesley R. Matheson, Stephen G. Mitchell, Talal Shamoon, Robert Endre Tarjan, and Francis Zane, Robustness and Security of Digital Watermarks, Springer-Verlag, Berlin, 1998, Proceedings of the Second International Conference on Financial Cryptography, Pages 227–240, 10.5555/647502.728328.

[10] Peter S. Pande, Robert P. Neuman, and Ronald R. Cavanagh, The Six Sigma Way: How GE, Motorola, and Other Top Companies are Honing Their Performance, 2000, McGraw-Hill.

[11] Vladimir D. Živković, Vyacheslav Shoshin, Abhijeet Golwelkar, and Ronald Peters, Towards Practical Traitor Tracing in Distributed and Large User-Bases, in Proceedings of the Mile-High Video Conference (MHV '26), Denver, CO, USA, Feb. 2–5, 2026. ACM, 2026. doi: 10.1145/3789239.3793291.

Author Biography

Vladimir D. Živković is Principal System Architect and Video Entertainment Innovation & Applied Research Lead at Irdeto. He holds a PhD in Computer Science from Leiden University. Since joining Irdeto in 2004, he has contributed to Cryptography & Security, Traitor Tracing, Ad Insertion, and Watermarking, and serves as lead architect for Irdeto's Forensic Watermarking Solutions.

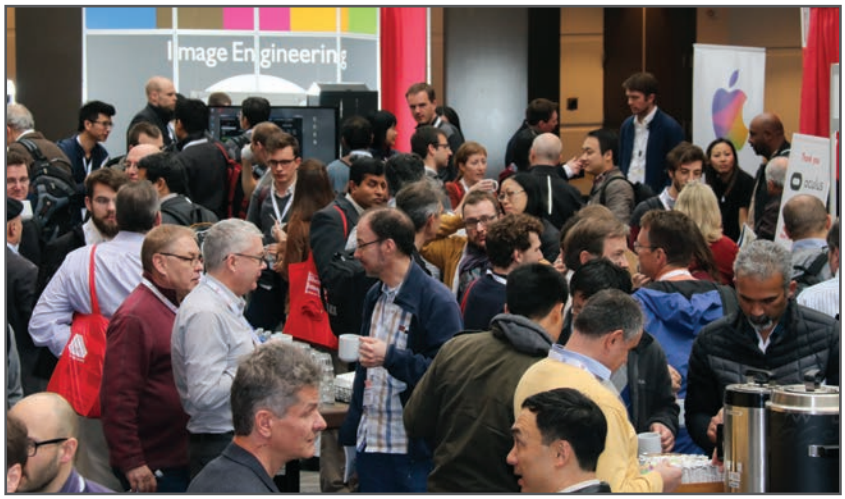
Abhijeet Golwelkar is Senior Solutions Architect at Irdeto. He holds a PhD in Electrical Engineering from Rensselaer Polytechnic Institute. He joined Irdeto in 2016 as a Software Engineer and since 2022, has been the main solutions contact in North America for Irdeto's forensic watermarking and anti-piracy products and services.

Ronald Peters is Senior Product Manager at Irdeto, leading Product Management for the TraceMark™Forensic Watermarking Portfolio since 2021. He holds a master's degree in Electronic Engineering and has experience in engineering and product management with companies including Nokia, Samsung, and Vodafone-Ziggo.

JOIN US AT THE NEXT EI!

electronic IMAGING

Imaging across applications . . . Where industry and academia meet!



• SHORT COURSES • EXHIBITS • DEMONSTRATION SESSION • PLENARY TALKS •
• INTERACTIVE PAPER SESSION • SPECIAL EVENTS • TECHNICAL SESSIONS •

www.electronicimaging.org

ADVANCING IMAGING SCIENCE 75 1947 IS&T 2022 imaging.org