

Security and Integrity of Audience Measurement Data in TV 3.0

João Pedro R. Norton

Eduardo Pagani Júlio

Marcelo F. Moreno

CITE THIS ARTICLE

Norton, João Pedro R.; Júlio, Eduardo Pagani; Moreno, Marcelo F.; 2026. Security and Integrity of Audience Measurement Data in TV 3.0. SET INTERNATIONAL JOURNAL OF BROADCAST ENGINEERING. ISSN Print: 2446-9246 ISSN Online: 2446-9432, doi: 10.18580/setijbe.2026.4 web link: <https://dx.doi.org/10.18580/setijbe.2026.4>



COPYRIGHT This work is made available under a Creative Commons Attribution-NonCommercial 4.0 International License.
Reproduction in whole or in part is permitted provided the source is acknowledged.

Security and Integrity of Audience Measurement Data in TV 3.0

João Pedro R. Norton, Eduardo Pagani Júlio and Marcelo F. Moreno, *Member, IEEE and SET.*

Abstract—Digital television systems increasingly collect audience measurement data directly from receivers, enabling granular and context-aware telemetry. The Brazilian TV 3.0 platform introduces a middleware-managed Audience Measurement Manager (AMM) that automates this through a standardized session lifecycle and structured reporting. Establishing trust in these reports is challenging because TV 3.0 receivers operate in a heterogeneous ecosystem without the user-account-based authentication common in digital platforms. This paper investigates security and integrity aspects of audience measurement data collection in TV 3.0, focusing on four threats: data interception, report-level tampering, bot-generated reports, and server overload. We examine cryptographic device attestation, using the Device Identifier Composition Engine (DICE) as a representative example, and instantiate the IETF RATS architecture as a technology-neutral trust chain involving the receiver, a verifier, the manufacturer, and the broadcaster, constrained by a mandatory offline baseline and preserving viewer privacy by exposing only a report hash and minimal evidence to the verifier.

Index Terms—TV 3.0, Audience Measurement Manager, Data Security, Data Integrity, Device Attestation, DICE, Remote Attestation, Digital Television

I. INTRODUCTION

BRAZILIAN digital television is experiencing a significant transformation with the advent of TV 3.0, a platform that succeeds previous generations, TV 2.0 and TV 2.5, which focused primarily on unidirectional broadcast delivery with no standardized return-channel communication at the platform level. TV 3.0 (a.k.a. DTV+) extends this foundation by considering broadband connectivity as a core platform requirement, enabling standardized bidirectional data flows between receivers and broadcaster infrastructure [1].

One of the primary benefits of this bidirectionality is the ability to automatically collect audience measurement data directly from viewer devices, through the Audience Measurement Manager (AMM), a middleware component responsible for orchestrating the entire data collection, session management, and report delivery process [4]. This capability positions TV 3.0 as one of the first broadcast systems to integrate audience measurement as a standardized, middleware-enforced platform service, rather than an application-level concern.

The data produced by the AMM has growing economic, statistical, and regulatory value. It underpins addressable advertising, proof-of-view mechanisms, audience metrics, and interactive campaigns, and increasingly informs commercial and regulatory decisions across the broadcast ecosystem. As

the value of this data grows, so does the incentive to manipulate it. This motivates the central concern of this work: raising the level of trust associated with the reports produced by DTV+ receivers, ensuring that collected data can be considered authentic and originating from legitimate devices.

In this model, the receiver collects information about content consumption and periodically reports it to servers configured by broadcasters. The confidentiality, authenticity, and integrity of the data transmitted through these channels are fundamental requirements for the collected information to be trustworthy and usable by third parties, such as audience research institutes.

However, ensuring these properties in an open broadcast television ecosystem presents challenges that do not exist in conventional authentication systems. Unlike web or mobile applications, TV receivers operate without user accounts, without login mechanisms, and without a centralized authentication server. Devices are manufactured by different vendors with heterogeneous hardware and software capabilities, and viewers have no credentials to present. This absence of prior identity creates opportunities for threats such as the generation of fraudulent reports by bots, interception of data in transit, and overloading of broadcaster servers with illegitimate requests.

Digital television systems in other regions, such as the North American ATSC 3.0 and the European HbbTV, protect the channel with TLS but do not specify mechanisms to prevent reports generated by illegitimate devices, leaving the device authentication problem unsolved [5], [6].

In response to this gap, this paper investigates security and integrity aspects of audience measurement data collection in TV 3.0, with particular emphasis on cryptographic device attestation. Rather than prescribing a single technology, we argue for a technology-neutral model in which only the cryptographic envelope, the minimal attestation evidence, and the data formats are standardized, while manufacturers remain free to choose the underlying attestation architecture best suited to their devices. As a concrete example of such an architecture, we examine the Device Identifier Composition Engine (DICE), which enables a receiver to prove its identity and firmware integrity without relying on an external certificate authority or dedicated security hardware. It is presented as one option among others rather than a mandatory requirement.

Building on this model, the paper instantiates the IETF Remote Attestation procedureS (RATS) architecture [7] in the AMM data flow, defining a trust chain involving the broadcaster, the receiver, an independent verifier, and the device manufacturer, with clearly separated responsibilities, and subject to the requirement that verification remain possible

Authors are with the Computer Science Department, Federal University of Juiz de Fora (UFJF), Juiz de Fora, MG, 36036-900 Brazil.

without contacting online services.

The remainder of this paper is organized as follows. Section II presents background concepts and related work. Section III describes the threat model and security requirements. Section IV analyzes the technical approaches investigated. Section V defines the trust chain and the roles of each participant. Section VI examines the trust properties of the model and the consequences of compromising each participant. Section VII presents the final remarks.

II. BACKGROUND AND RELATED WORK

A. TV 3.0 Audience Measurement

The Audience Measurement Manager (AMM) is a middleware component of the TV 3.0 Application-oriented Platform, responsible for orchestrating audience data collection, session management, and report delivery [4]. Audience measurement is initiated when a broadcaster transmits, via broadcast channel, a manifest containing campaign configuration parameters, including the URL for signature and URLs to which collected data must be delivered. Upon viewer agreement, the AMM begins collecting behavioral and contextual data, which is periodically reported to broadcaster-defined endpoints.

Each audience measurement session is uniquely identified by the pair (Service@globalServiceID, viewer-id), binding the collected data to both the broadcast service and the active viewer profile. Two URLs are central to the data flow: one for delivering audience measurement reports (deliveryURLs) and one for obtaining digital signatures over those reports (signatureRetrievalURL). Ensuring the security and integrity of communications through these endpoints is the central concern of this paper.

B. Related Standards: ATSC 3.0 and HbbTV

The North American ATSC 3.0 system addresses channel security through TLS 1.2 and 1.3 for transport protection, and adopts the Cryptographic Message Syntax (CMS) standard for data signing [5]. The European HbbTV standard similarly mandates TLS 1.2 and 1.3 for secure communication [6]. While both systems establish a baseline for confidentiality and data integrity, neither specifies mechanisms to prevent the generation of reports by illegitimate devices or bots. Device authentication is left unaddressed, and no standard return-path trust model is defined at the platform level.

C. Machine-to-Machine Authentication

Authenticating devices without prior user identity is a well-known challenge in Machine-to-Machine (M2M) and Internet of Things (IoT) ecosystems. The OAuth 2.0 Client Credentials flow [8] is widely used in M2M contexts, where a device holds a client_id and client_secret and exchanges them for a signed access token issued by a central authorization server. However, this model is not directly applicable to TV 3.0, as it requires a centralized authentication infrastructure and assumes that device credentials are provisioned and maintained by manufacturers, conditions that cannot be guaranteed across the heterogeneous TV 3.0 device ecosystem.

IoT platforms such as AWS IoT [14] and Microsoft Azure IoT Hub [15] address device authentication through X.509 certificates and mutual TLS (mTLS), where each device holds a unique certificate issued during manufacturing. While effective, these approaches also depend on manufacturer involvement for certificate provisioning and revocation, and assume a degree of hardware security that may not be present in all TV 3.0 receivers. Smart TV platforms such as Samsung Tizen expose device unique identifiers (DUIDs) through proprietary APIs [16], but these are platform-specific and unavailable on dedicated DTV+ receivers, making them unsuitable as a universal solution.

D. Remote Attestation Architecture

The problem of establishing whether a remote device is in an expected operating state has been addressed architecturally by the IETF in the Remote Attestation procedureS (RATS) architecture [7]. Rather than defining a protocol or a data format, the document establishes a vocabulary and a set of roles intended to be mapped onto existing and emerging attestation procedures, so that solutions developed independently can be described in comparable terms.

Four roles are central to the architecture. The Attester produces Evidence about itself. The Verifier appraises that Evidence against an appraisal policy and produces Attestation Results. The Relying Party consumes Attestation Results and applies its own policy to make application-specific decisions. The Endorser, typically the manufacturer, produces Endorsements, which are secure statements vouching for the integrity of the capabilities of the Attester, such as its ability to collect claims and to sign Evidence. A fifth role, the Reference Value Provider, supplies the values against which claims in the Evidence are compared. A single entity may aggregate more than one role, and the Endorser and Reference Value Provider roles are commonly held by the same manufacturer.

The architecture also describes two topological patterns for the flow of these messages. In the Passport Model, the Attester conveys Evidence to the Verifier, receives an Attestation Result in return, and later presents that result to the Relying Party, by analogy with a citizen who obtains a passport from an issuing authority and presents it at an immigration desk. In the Background-Check Model, the Attester conveys Evidence to the Relying Party, which treats it as opaque and forwards it to a Verifier, then acts on the Attestation Result it receives back. The two patterns are presented as reference models and do not constrain the interactions between roles.

Trust in this architecture is expressed through trust anchor stores. A Relying Party trusts a Verifier by holding its key material, and a Verifier comes to trust an Attester indirectly by holding the key material of an Endorser that vouches for it, rather than by knowing each device individually.

DICE, described in the following subsection, operates on the Attester side of this architecture. It is a mechanism by which a device produces Evidence about its identity and software state, and it is independent of the topological pattern in which that Evidence is later appraised.

E. Device Identifier Composition Engine (DICE)

The Device Identifier Composition Engine (DICE) is a hardware-rooted mechanism developed by the Trusted Computing Group (TCG) that enables a device to prove its identity and the integrity of its firmware without relying on a Trusted Platform Module (TPM) or an external certificate authority [10]. It has been adopted into Android as part of its security architecture [11], demonstrating its practical applicability to large-scale, heterogeneous device ecosystems. It is presented here not as the proposed solution, but as a representative example of an attestation architecture that is particularly well suited to constrained devices such as IoT systems and low-cost TV receivers, where dedicated security hardware may not be available. As discussed in Section IV, the model proposed in this paper is technology-neutral and does not mandate DICE specifically.

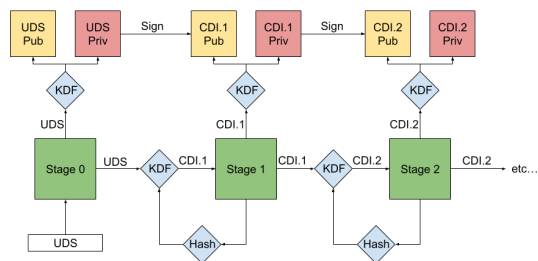


Fig. 1. Simplified DICE derivation process. Source: [11]

As illustrated in Fig. 1, the DICE process begins with a Unique Device Secret (UDS), a statistically unique value stored in protected read-only memory accessible only to the DICE layer. At each boot stage, DICE measures the firmware to be loaded, producing a TCB Component Identifier (TCI) — a cryptographic hash of the firmware image and configuration. The UDS and TCI are then combined using a key derivation function (KDF), a one-way cryptographic function that derives one or more secret keys from a secret input, to produce a Compound Device Identifier (CDI), which serves as the cryptographic identity of the device for that boot stage. This process is chained across successive boot layers, so that any modification to firmware at any stage produces a different CDI, invalidating all keys derived from it. From the final CDI, the device derives asymmetric key pairs used to sign attestation evidence, allowing a verifier to confirm both the identity and the software state of the device without direct access to the UDS.

III. THREAT MODEL AND SECURITY REQUIREMENTS

This section identifies the principal security threats affecting the audience measurement data flow in TV 3.0 and derives the corresponding security requirements that any proposed solution must address.

Audience measurement reports are transmitted over broadband networks from receiver to broadcaster servers. Without adequate transport protection, an adversary positioned between the receiver and the server could intercept these communications, exposing viewer behavioral data and potentially allowing

report manipulation. Confidentiality and integrity of data in transit are therefore fundamental requirements. Both ATSC 3.0 and HbbTV address this threat through TLS 1.2 and 1.3 [5], [6], and the same approach is applicable to TV 3.0.

Beyond transport security, the broadcaster must be able to verify that a received report was generated by a legitimate receiver and has not been tampered with after collection. This requires mechanisms for data origin authentication and integrity protection at the report level, independent of the transport channel. A report could be intercepted, modified, and retransmitted even over a TLS-protected channel if the report itself carries no cryptographic proof of origin. This threat motivates the use of digital signatures over the report payload, as specified in the TV 3.0 AMM through ECDSA/SHA-256 [4].

The most critical threat in this context is the possibility of fraudulent reports generated by automated bots or software emulators that mimic the behavior of real receivers without being legitimate TV 3.0 devices. Unlike the previous threats, this one cannot be addressed by transport security or report-level signatures alone, since a bot could establish a valid TLS connection and produce a correctly signed report if it has access to valid credentials. Preventing this class of attack requires mechanisms capable of binding a report to a specific physical device, establishing that the data originated from a real receiver running legitimate software. This is the central security challenge addressed by this paper, and it motivates the attestation based trust chain described in the following sections.

Server overload through illegitimate requests represents an additional operational concern. An attacker could flood broadcaster delivery endpoints with a large volume of fabricated reports, degrading service availability for legitimate receivers. While classical rate limiting offers only partial mitigation against distributed attacks [9], device-level attestation contributes to mitigating this threat, since requests carrying unrecognized or invalid device evidence can be rejected early in the processing pipeline, reducing the cost of handling illegitimate traffic.

IV. INVESTIGATED TECHNICAL APPROACHES

This section surveys the technical approaches investigated to address the security requirements identified in Section III, discussing their applicability and limitations in the TV 3.0 context.

Transport Layer Security (TLS) is the natural starting point for protecting data in transit. Both ATSC 3.0 and HbbTV adopt TLS 1.2 and 1.3 for channel protection [5], [6], and the same applies to TV 3.0. Mutual TLS (mTLS) extends this model by requiring the client to present a certificate during the handshake, effectively authenticating the device at the transport layer. While mTLS would partially address the bot threat, it requires each device to hold a certificate issued by a trusted authority, which depends on manufacturer provisioning that cannot be guaranteed across the heterogeneous TV 3.0 device ecosystem.

JSON Web Tokens (JWT) [17] combined with X.509 certificates [12] represent another candidate approach for report-

level authentication. In this model, the receiver would sign the report payload using a private key, embedding the corresponding X.509 certificate in the JWT header. The broadcaster could then verify the signature and validate the certificate chain. However, this approach faces the same fundamental limitation as mTLS: it requires devices to hold certificates issued by a trusted authority, and the cost and complexity of maintaining a public key infrastructure (PKI) at the scale of the TV 3.0 ecosystem, potentially involving a Brazilian authority such as ICP-Brasil, was identified as a significant practical barrier.

Trusted Execution Environments (TEE) and Trusted Platform Modules (TPM) were also considered as candidates for secure key storage and cryptographic operations on the device. A TEE provides an isolated execution environment within the main processor, allowing sensitive operations such as key generation and signing to be performed without exposure to the main operating system [13]. TPMs offer similar guarantees through dedicated hardware security chips. Both mechanisms are widely used in PC and mobile platforms and would provide strong security guarantees, if available. However, their presence cannot be assumed across the full spectrum of TV 3.0 receivers, which includes low-cost set-top boxes and dedicated DTV+ devices from multiple manufacturers. The absence of a standardized TEE or TPM requirement in the TV 3.0 specification makes these approaches unsuitable as a universal solution.

The Device Identifier Composition Engine (DICE), described in Section II-E, illustrates how cryptographic device attestation can be achieved without dedicated security hardware. In the TV 3.0 context, a verifier could confirm that a report was signed by a specific device running unmodified firmware, providing strong guarantees against bot-generated reports. DICE is particularly attractive for low-cost receivers, since it can reuse the Secure Boot mechanisms already present in many devices and avoids the cost of mandatory dedicated hardware. Its main limitation is that DICE support must be present in the device hardware and firmware, which depends on manufacturer decisions.

Rather than mandating any single one of these technologies, the model proposed in this paper is technology neutral. None of TPM, TEE, or DICE is required; instead, only the cryptographic envelope, the minimal attestation evidence, and the data formats exchanged with the verifier are standardized. This approach preserves interoperability across manufacturers while allowing each vendor to adopt the attestation architecture best suited to its devices, from full hardware implementations to firmware-based solutions such as DICE on constrained receivers. The roles and interactions that make this model operational are described in the following section.

V. TRUST CHAIN AND RESPONSIBILITIES

The trust model proposed in this paper is built upon the data flow already defined in ABNT NBR 25608 [3], requiring no structural changes to the existing AMM architecture. The receiver generates the report through the AMM, sends a request to the signatureRetrievalURL, receives a signature or authorization token in response, and delivers the report to the

configured deliveryURLs. The proposal adds only an operational attestation envelope and the associated cryptographic evidence to this flow. Four participants take part in the resulting trust chain: the receiver, the verifier, the manufacturer, and the broadcaster or its partner.

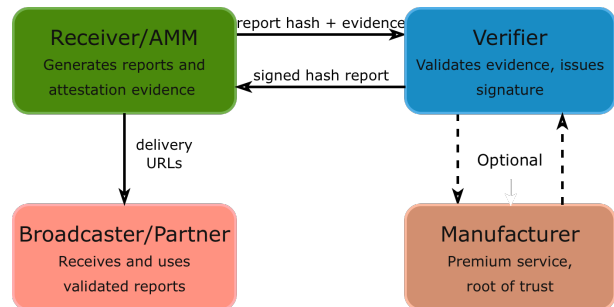


Fig. 2. Trust chain and roles in the proposed attestation model.

As summarized in Fig. 2, each participant has a clearly delimited role, which maps onto the roles of the RATS architecture introduced in Section II-D. The receiver, through the AMM, generates the report and produces the cryptographic evidence that attests to its origin and integrity, acting as the Attester. The verifier, reached through the signatureRetrievalURL, appraises this evidence and issues the corresponding signature or authorization, acting as the Verifier. The manufacturer aggregates the Endorser and Reference Value Provider roles, supplying the root of trust for device identity, the firmware measurements expected of compliant devices and, optionally, additional verification policies. The broadcaster or its partner is the Relying Party, receiving the validated reports and using them for audience measurement and related purposes. A central property of this separation is that the manufacturer never receives telemetry or viewer behavior, and the verifier never accesses the report content, ensuring that no single party concentrates both the identity and the behavioral data of the viewer.

The flow follows the Passport Model: the receiver obtains the attestation result before delivery and presents it to the broadcaster, rather than having the broadcaster forward evidence to a verifier on its behalf. This pattern was not chosen for conformance but because it suits the constraints of the domain. It keeps the verifier off the delivery path and therefore out of the report traffic, it allows a single verification to serve all of the configured deliveryURLs, and it leaves the broadcaster independent of verifier availability at the moment a report arrives. What follows is an instantiation of this architecture under two constraints that the generic architecture does not impose: verification must remain possible without contacting online services, and the verifier must never observe report content.

Because the verifier signs a hash rather than the report itself, the binding between the two must be closed at the point of delivery. Upon receiving a report, the broadcaster or its partner recomputes the hash over the delivered content and checks it against the hash covered by the verifier signature. A report whose recomputed hash does not match the signed one is

rejected, since the signature attests to a different payload. Without this check the two halves of the flow would be independent: a receiver could obtain a valid signature over one report and deliver another, and the attestation would then guarantee only that some report had been validated, not the one actually received. This step is what allows the verifier to remain unaware of report content while the attestation evidence stays bound to a specific payload.

The interactions described above presuppose that each participant already holds a set of trust anchors. The receiver holds the key material and the certificate chain provisioned during manufacturing, which it cannot obtain later on its own. The verifier holds the set of manufacturer roots it recognizes and the reference values against which it compares firmware measurements, together with the appraisal policy it applies to evidence derived from them. These are the Endorsements and Reference Values of the architecture described in Section II-D, and they reach the verifier out of band rather than through the flow above. The broadcaster or its partner holds the public key of the verifier whose signatures it accepts. None of these anchors is established by the protocol itself; they are inputs to it.

This displaces the trust problem rather than removing it. The guarantees of the chain are bounded by the decision to admit a given manufacturer root into the trusted set of the verifier: evidence that validates correctly against the root of a manufacturer that signs arbitrary values is indistinguishable, at the cryptographic level, from evidence produced by a compliant device. Establishing the criteria and the process by which manufacturer roots are recognized is therefore a precondition for the model rather than a deployment detail, and it is a question of who may act as Endorser rather than one that cryptography can settle.

A mandatory offline baseline ensures interoperability and operational independence. Every receiver transmits, as part of its report, a cryptographic signature, the trust chain linking its keys to the manufacturer root, and the firmware measurements that characterize its software state. This baseline allows the verifier to authenticate a report and confirm the integrity of the originating device using only the evidence carried in the report itself, without depending on permanently available online services from the manufacturer. As a result, the core verification mechanism remains functional even when external services are unavailable, which is essential for a broadcast ecosystem expected to operate at national scale.

On top of this baseline, an optional hybrid validation mode enables stronger and more dynamic guarantees. While the verifier always performs offline validation, it may additionally consult the manufacturer to check for key revocation, minimum acceptable firmware versions, device reputation, and antifraud policies. This hybrid consultation does not alter the protocol, the attestation envelope, or the report format; it complements the offline baseline with operational and dynamic antifraud capabilities. This optional layer also opens the possibility for manufacturers to offer premium verification services, providing faster responses and more up to date validation data to broadcasters or verifiers that require higher assurance levels, while leaving the basic functionality available to all

participants at no additional dependency.

Privacy preservation is an explicit design goal of this model. The signatureRetrievalURL receives only a hash of the report together with minimal cryptographic evidence, never the report content itself. The verifier therefore has no access to the viewer-id, the deliveryURLs, or any behavioral data, and is unable to reconstruct what the viewer watched or how the receiver was used. This separation significantly reduces the risks of undue correlation, traceability, and regulatory exposure, aligning the architecture with data protection requirements such as Brazil's Lei Geral de Proteção de Dados (LGPD) [2], while still allowing independent verification of report authenticity and integrity.

VI. TRUST PROPERTIES AND RESIDUAL RISK

This section examines two properties of the model that follow from the separation of roles described above: how verification outcomes reflect heterogeneous attestation strength, and what an adversary obtains from compromising each participant. Unlike Section III, which considers threats to the data flow posed by an external adversary, the analysis here concerns the failure of the participants themselves.

A model that does not mandate a specific attestation technology necessarily admits evidence of unequal strength. A receiver whose keys are held in a dedicated security module, one whose identity is derived from firmware measurements taken at boot, and one that offers no hardware-rooted evidence at all can all participate in the flow described above, but the confidence that a validated report originated from a genuine device differs substantially across these cases. Treating the outcome of verification as a binary decision would erase that difference, presenting reports backed by very different guarantees as equivalent to the party that consumes them. This is a general property of attestation rather than a peculiarity of TV 3.0: attesters differ in strength of function, and attestation results are correspondingly different in strength even when appraisal succeeds in every case [7].

The verifier is therefore better understood as issuing a qualified rather than an absolute statement. Alongside the signature over the report hash, it declares the level of assurance that the evidence supports, derived from the class of attestation the device provided and from the appraisal criteria applied to it. The broadcaster or its partner, which alone knows the purpose the data will serve, then applies its own policy to that declaration. A campaign used for internal audience estimates may accept every validated report regardless of level, while one that underpins contractual proof of view may require a minimum level, or may segregate reports by level so that the composition of the sample remains visible in the analysis.

This preserves technology neutrality without asserting that all implementations are equally trustworthy: the consequences of each manufacturer's choice become visible to the parties that depend on it rather than being absorbed silently by the verifier, giving vendors consistently assigned a low level a commercial reason to strengthen their implementation.

The value of separating roles becomes clearer when each participant is considered as a potential point of failure. An

adversary controlling the verifier can issue valid signatures over arbitrary report hashes, and fabricated reports would then be accepted by every broadcaster that trusts it. The damage is confined to integrity, since the verifier holds no report content or viewer identifiers from which a viewer profile could be reconstructed. Anomalies in issuance volume relative to the plausible installed base offer a means of detection independent of the cryptographic layer.

Compromise of a manufacturer root allows an adversary to mint device identities for that manufacturer and to produce evidence that validates correctly, indistinguishable from legitimate evidence at the cryptographic level. The consequence is bounded to the device population associated with that root, and roots belonging to other manufacturers remain unaffected. This containment is an argument for a plurality of manufacturer roots rather than a single national root, whose compromise would invalidate the guarantees of the entire ecosystem at once. Compromise of the optional manufacturer service used in hybrid mode is more limited still. Because the offline baseline is mandatory and the consultation only complements it, an adversary in that position can suppress revocation notices or accept outdated firmware versions, but cannot cause a report to be authenticated that offline validation would have rejected. The dynamic guarantees degrade to those of the baseline rather than falling below it.

Extraction of the keys of an individual receiver allows an adversary to produce valid reports under that device identity. The attack is bounded to a single identity, and its usefulness for large-scale fraud is limited by the volume of reports a single receiver can plausibly emit before becoming statistically conspicuous. Because firmware measurements form part of the evidence, modifying the software of the device produces different measurements and therefore a distinguishable identity, so the attack requires extracting keys without altering the firmware that protects them.

Compromise of the broadcaster or its partner lies outside what attestation can address. These parties are the intended recipients of the data and may alter it after receipt; the model establishes that a report was produced by a genuine device, not that the party consuming it reports honestly. Guarantees at that level would require auditing arrangements external to the mechanisms discussed here.

Across these cases one property holds throughout: no single compromise yields both the identity of a device and the behavior of its viewer. The manufacturer never receives telemetry, and the verifier never receives report content, so the correlation most damaging to viewers requires collusion between parties that the architecture keeps apart.

VII. FINAL REMARKS

This paper investigated security and integrity aspects of audience measurement data collection in TV 3.0. Motivated by the growing economic, statistical, and regulatory value of AMM data and by the device authentication gap left open by ATSC 3.0 and HbbTV, we surveyed approaches such as TLS, mTLS, JWT, X.509, TEE, and TPM, and examined DICE as a representative example of attestation without dedicated security hardware. On this basis, we instantiated the IETF RATS

architecture as a technology-neutral trust chain involving the receiver, an independent verifier, the manufacturer, and the broadcaster, following the Passport Model and anchored by a mandatory offline baseline, complemented by an optional hybrid validation mode that opens the possibility of premium manufacturer services. A central property of the design is privacy preservation: the verifier processes only a report hash and minimal cryptographic evidence, never the report content, viewer identifiers, or behavioral data, aligning the architecture with privacy protection regulatory frameworks.

Several directions remain open for future work: integrating attestation mechanisms such as DICE into TV 3.0 receiver specifications, in coordination with manufacturers; a formal security analysis of the trust chain against replay, forgery, and spoofing attacks; and the standardization of the attestation envelope and evidence formats toward a deployable specification.

REFERENCES

- [1] Fórum SBTVD, "TV 3.0 Project," https://forumsbtvd.org.br/tv3_0/, accessed May 2026.
- [2] Brasil, *Lei n.º 13.709, de 14 de agosto de 2018. Lei Geral de Proteção de Dados Pessoais (LGPD)*. Diário Oficial da União, Brasília, DF, 2018.
- [3] Associação Brasileira de Normas Técnicas, *ABNT NBR 25608: TV 3.0 – Application Coding*. ABNT, Brazil, 2025.
- [4] M. F. Moreno and E. Barrère, "TV 3.0 Audience Measurement Management: Architecture, Lifecycle and APIs," *SET International Journal of Broadcast Engineering*, vol. 11, 2025, doi: <https://doi.org/10.18580/setijbe.2025.5>.
- [5] Advanced Television Systems Committee, *ATSC A/360:2024-04 Security*. Washington, DC, USA: ATSC, 2024.
- [6] European Telecommunications Standards Institute, *Hybrid Broadcast Broadband TV (ETSI TS 102 796 V1.7.1)*. Sophia Antipolis, France: ETSI, 2023.
- [7] Internet Engineering Task Force, *RFC 9334: Remote Attestation procedureS (RATS) Architecture*. IETF, 2023, <https://www.rfc-editor.org/rfc/rfc9334>.
- [8] Internet Engineering Task Force, *RFC 6749: The OAuth 2.0 Authorization Framework*. IETF, 2012, <https://www.rfc-editor.org/rfc/rfc6749>.
- [9] S. T. Zargar, J. Joshi, and D. Tipper, "A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 2046–2069, 4th Quart., 2013, doi: <https://doi.org/10.1109/SURV.2013.031413.00127>.
- [10] Trusted Computing Group, *DICE Attestation Architecture, Version 1.2*. TCG, 2021, https://trustedcomputinggroup.org/wp-content/uploads/DICE-Attestation-Architecture-v1.2_pub.pdf.
- [11] Google, "Device Identifier Composition Engine," Android Open Source Project, <https://source.android.com/docs/security/features/dice>, accessed May 2026.
- [12] Internet Engineering Task Force, *RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. IETF, 2008, <https://www.rfc-editor.org/rfc/rfc5280>.
- [13] M. Sabt, M. Achemlal, and A. Bouabdallah, "Trusted Execution Environment: What It is, and What It is Not," in *Proc. 2015 IEEE Trustcom/BigDataSE/ISPA*, Helsinki, Finland, 2015, vol. 1, pp. 57–64, doi: <https://doi.org/10.1109/Trustcom.2015.357>.
- [14] Amazon Web Services, "Authorizing Direct Calls to AWS Services Using AWS IoT Core Credential Provider," <https://docs.aws.amazon.com/iot/latest/developerguide/authorizing-direct-aws.html>, accessed May 2026.
- [15] Microsoft, "Authenticate to Azure IoT Hub Using SAS Tokens," <https://learn.microsoft.com/en-us/azure/iot-hub/authenticate-authorize-sas>, accessed May 2026.
- [16] Samsung, "ProductInfo API - getDuid," <https://developer.samsung.com/smarttv/develop/api-references/samsung-product-api-references/productinfo-api.html>, accessed May 2026.
- [17] Internet Engineering Task Force, *RFC 7519: JSON Web Token (JWT)*. IETF, 2015, <https://www.rfc-editor.org/rfc/rfc7519>.



contributes to the development of the TV 3.0 test suite. This work was developed as part of his undergraduate thesis under the TV 3.0 Project.

João Pedro R. Norton received the B.S. degree in Exact Sciences from the Federal University of Juiz de Fora (UFJF), Juiz de Fora, Brazil, in 2026, where he is currently pursuing the B.S. degree in Computer Science. He is an intern at the RePesq data center, where he works with the infrastructure of an integrated high-speed research network. He also works as a backend developer at Atlantis, implementing the authentication system of the DTV+ Common Platform. He is a research fellow at RNP (Rede Nacional de Ensino e Pesquisa), where he



Eduardo Pagani Júlio is an Associate Professor at the Federal University of Juiz de Fora (UFJF), Brazil. He holds a Ph.D. in Computer Science from Fluminense Federal University (UFF) and works in the fields of computer networks and cybersecurity. His current research focuses on audience measurement and privacy-aware media platforms for next-generation digital television within the Brazilian Digital TV System Forum (SBTVD).



the Brazilian Digital TV System Forum (SBTVD) and serves as editor of the ABNT standards for TV 2.5 and TV 3.0 on middleware and application coding. His research focuses on application-oriented broadcasting, second-screen integration, audience measurement, and privacy-aware media platforms for next-generation digital television. He has published widely in the field and is a member of the Brazilian Society of Television Engineering (SET), the Institute of Electrical and Electronics Engineers (IEEE) and the Brazilian Computer Society (SBC).

Marcelo F. Moreno is an Associate Professor at the Federal University of Juiz de Fora (UFJF), Brazil. He holds a Ph.D. in Computer Science from PUC-Rio and works in the areas of multimedia systems and computer networks. From 2022 to 2023, he was a Visiting Professor at the International Audio Laboratories Erlangen (FAU/Fraunhofer IIS), Germany. He co-edited several ITU-T and ITU-R Recommendations and Technical Reports, coordinating ITU-T working groups for over a decade. Since 2015, he has led the Application Coding Working Group of