August 2026

Digital Out-of-Home Advertising Cyber Security Guideline

Cyber Controls and Implementation Guidance

1 Contents

1 Contents 2
2 Version Control 3
3 Background 4
4 How Controls are Organised 6
5 Controls 7
6 Appendix – Asset Definitions 34
7 Appendix – Third Party Attestation Questionnaire 37
8 Appendix – Technical Security Assessment Scopes 44

2 Version Control

Version	Changes	Author	Date
0.1	Creation of document	Skylight Cyber	1/07/2026
0.11	Feedback from OMA	OMA	6/07/2026
1.0	Addressed feedback and finalised	Skylight Cyber	5/08/2026

3 Background

The DOOH ecosystem has evolved from relatively static advertising workflows into a digital, software-driven supply chain. Campaigns may now involve dynamic creative optimisation, third-party verification, programmatic trading, live data feeds, content delivery networks, CMS platforms, and player devices that receive, store, fetch, and execute content across many environments. This complexity makes it harder for individual participants to know which controls are expected of them, where their responsibilities start and end, and how to assess whether their current practices are appropriate for the risks they manage.

The Digital Out-of-Home Advertising Cyber Security Guideline (referred to as the 'Guideline') provide practical guidance for applying the Digital Out-of-Home Advertising Cyber Security Framework (referred to as the 'Framework'). While the Framework defines the industry's shared security outcomes and responsibilities, this Guideline explain what good looks like in practice for the organisations and suppliers involved in creating, transferring, approving, scheduling, and rendering DOOH content.

This Guideline is intended to help the industry interpret the Framework consistently and implement it in a way that is proportionate, practical, and specific to DOOH. It sets out recommended controls, implementation considerations, and examples of good practice where relevant, so organisations can assess their current posture, identify gaps, and improve security across the content delivery chain without having to translate generic cyber security standards into the DOOH context from first principles.

The purpose of the combined documents is to collectively enhance and manage the cyber risk to security and subsequent revenue across the DOOH industry while continuing to provide a safe and trusted environment for advertisers.

3.1 Who is this for

This Guideline is written for OMA member organisations and participants across the broader DOOH supply chain that need to understand and apply the Framework in practice. This includes media owners, advertisers, media and creative agencies, DCO providers, verifiers, SSPs, DSPs, CMS and player software vendors, and other suppliers involved in creating, transferring, approving, scheduling, delivering, or rendering DOOH content. The audience includes business, operational, technology, security, risk, procurement, and supplier management teams. Readers are expected to understand their role in the DOOH ecosystem, but do not need to be cyber security specialists.

3.2 Structure of this Guideline

Section 5 (Controls) is the operational core of this document. It is organised into control families, each of which corresponds to a security sub-outcome from the Framework. Every control family follows the same structure:

- An overview explaining what the family addresses; and
- The sub-controls, each assigned an implementation tier (Minimum, Recommended, or Target), referencing the assets it applies to, and accompanied by DOOH-specific implementation guidance where relevant.

The appendices provide supporting reference material for this Guideline. This includes sample technical security assessment scopes tailored to the key participant roles in the DOOH supply chain, and (updated) third-party attestation questionnaires for supply chain participants.

3.3 Scope

This Guideline provides implementation guidance for the four cyber security outcomes defined in the Digital Out-of-Home Advertising Cyber Security Framework: Protect Content Integrity, Preserve Delivery Chain Integrity, Secure the Player Environment, and Prevent Industry-Wide Compromise.

This Guideline is focused on areas where the DOOH industry has specific operating models, technical dependencies, supply chain relationships, or risk considerations that require interpretation beyond general cyber security practice. This includes the handling of creatives and content packages, approval and transfer workflows, programmatic and dynamic content pathways, verifier and platform integrations, CMS and player software environments, player device infrastructure, and the shared supplier platforms on which multiple participants may depend.

This Guideline is not intended to replace established cyber security standards or duplicate controls that are already well covered by existing best practice. Where suitable guidance already exists, organisations should apply those standards and use this Guideline to understand how they relate to DOOH-specific workflows and responsibilities. The controls and implementation considerations in this document are therefore scoped to the Framework outcomes and emphasise the points where DOOH participants need practical guidance on what good looks like to deliver safe practice for this industry.

4 How Controls are Organised

4.1 Control family

Controls in this Guideline are organised into control families. Each control family corresponds to one of the security sub-outcomes defined in the Framework, for example 'Execution Integrity' or 'Custody Preservation'. A control family groups together all the practical measures that address a particular security objective.

4.2 Sub-control

Within each control family, controls are broken down into sub-controls. A sub-control is a specific, actionable measure that contributes to the control family's objective. Sub-controls describe what a participant should do or have in place, not how to configure a specific product or system, so they apply across the diverse technology environments in the DOOH industry.

4.3 Implementation guidance

Some sub-controls include implementation guidance. This guidance is included where DOOH-specific context is needed to apply the control effectively, and where existing best practice does not adequately address the operating model, technical environment, or supply chain relationships involved. It is not repeated where general cyber security standards already provide sufficient direction.

Implementation guidance may cover how a control applies differently across delivery pathways, what good looks like for a particular content type or participant role, considerations specific to the programmatic environment or dynamic content workflows, or how to apply a control where the DOOH industry's shared supplier model creates dependencies that general guidance does not anticipate.

4.4 Implementation tiers

Each sub-control is assigned one of three implementation tiers that indicate the level of maturity expected.

- **Minimum:** The baseline expected of all participants in the relevant role. These controls address the highest-likelihood risks and should be in place regardless of organisation size or technical maturity.
- **Recommended:** Good practice that most participants should be working toward. These controls improve resilience and reduce risk meaningfully but may require more investment or operational change than the minimum baseline.
- **Target:** Leading practice for participants with higher risk exposure, greater technical capability, or where the consequences of compromise are most significant.

As higher-risk technologies such as dynamic content are adopted and the threat landscape evolves, implementation tiers may change. For example, a control rated 'Recommended' today may become 'Minimum' in future.

5 Controls

5.1 Overview

The following controls are organised into ten Control Families (CF) covering the points at which Digital OOH content is created, transferred, approved, scheduled, delivered and displayed. Each family contains sub-controls assigned to a tier so organisations can distinguish the minimum baseline from recommended uplift and target-state controls for higher-risk dynamic content pathways.

Control family	Outcome Mapping	Minimum	Recommended	Target	Total Sub-Controls
CF01 - Content Approval and Inspection	1.1, 2.3	6	3	2	11
CF02 - Execution Integrity and Runtime Bounding	1.2, 1.3, 1.4	-	8	1	9
CF03 - Runtime Monitoring and Detection	1.5	-	-	2	2
CF04 - Origin to Destination Authenticity	2.1	-	-	2	2
CF05 - Custody and Transfer Protection	2.2	1	1	-	2
CF06 - Player Physical Integrity	3.1	2	2	-	4
CF07 - Player and Network Segregation	3.2	2	2	1	5
CF08 - Identity, Access and Accountability	3.3	4	5	2	11
CF09 - Supply-chain Assurance and Blast-radius Control	4.1, 4.2	4	3	1	8
CF10 - Baseline System and Operational Hygiene	Baseline	6	3	-	9
Total Sub-Controls	-	**25**	**27**	**11**	**63**

5.2 CF01 - Content Approval and Inspection

5.2.1 Overview

This control family governs the assessment of creative content before it is approved for scheduling or delivery. Content packages are classified by type, which determines the depth of review and the attestation required of submitting parties. Pre-approval review is supported by capability manifests, sandboxed rendering, and code inspectability requirements. Higher-assurance tiers add a production-matched review environment and extended pre-approval testing periods.

5.2.2 CF01-01 - Content package classification

Type	Applies To	Responsible Party	Enabling Party
Minimum	Content package	Media owner	-

5.2.2.1 Requirement

The receiving party defines and documents a classification for content - fully self-contained, semi self-contained, or remote - and determines the expected classification for each submitting party's content. The classification is the first step in protecting content and governs what follows, because it determines the security risk. The receiving party uses it to determine, for each submitting party and content type: the protective controls that apply, the depth of pre-approval review, and the attestation requirements placed on the submitting party.

5.2.2.2 Implementation guidance

Classify content at the level of the submitting party: fully self-contained (all visible and non-visible assets are included in the package; the package makes no network requests while it runs), semi self-contained (all visible content is included in the package; only non-visible assets, such as verification scripts or measurement tags, are loaded from approved remote locations), or remote package (visible content is loaded from approved remote locations, such as fully dynamic creative content). If a submitting party provides more than one content type, all relevant content type requirements should be combined to determine the most restrictive combination of controls.

5.2.3 CF01-02 - Definition of content package controls

Type	Applies To	Responsible Party	Enabling Party
Minimum	Content package	Media owner	-

5.2.3.1 Requirement

Based on the classification of the content, the receiving party selects and documents the set of protective and detective controls that apply to each content type, commensurate with its risk. The selected controls form the agreed control baseline for that content type and inform both the attestation required of the submitting party and the depth of pre-approval review.

5.2.3.2 Implementation guidance

Use the classification as the basis for control selection. For each content type, document the controls that apply and the party responsible for each, so that submitting parties know what is expected of their content and reviewers know what to verify. Refer to the following Control Families to select the appropriate controls: CF02 - Execution integrity and runtime bounding, CF03 - Runtime monitoring and detection and CF04 – Origin Authenticity.

5.2.4 CF01-03 - Definition of attestation requirements

Type	Applies To	Responsible Party	Enabling Party
Minimum	Content package	Any party that receives a package	-

5.2.4.1 Requirement

The receiving party defines the content and scope of the conformance attestation it requires from submitting parties and communicates these to those parties.

5.2.4.2 Implementation guidance

The receiving party is any party that accepts content or code from another into its environment. It documents the set of statements that comprise the conformance attestation and, for each, the requirement to which it relates. This defined attestation constitutes the agreed basis referenced by the submitting-party conformance attestation. The receiving party determines the submitting parties to which the attestation applies, the point in the acceptance, submission or scheduling workflow at which it is captured, and the record to be retained. The defined attestation is reviewed and updated as the receiving party's requirements change, and material changes are communicated to the affected parties. Where the attestation is required of external parties, the obligation is reflected in the relevant agreements.

5.2.5 CF01-04 - Submitting-party conformance attestation

Type	Applies To	Responsible Party	Enabling Party
Minimum	Content package	DCO agency, Verification provider	-

5.2.5.1 Requirement

When content or code is submitted to another party, the submitting party makes a formal, attributable declaration that it conforms to the standards and guidelines agreed with the receiving party and contains no known malicious code. The declaration is recorded against the submitted item and the named individual or organisation, creating a documented point of individual accountability.

5.2.5.2 Implementation guidance

The declaration is captured as a defined step within the submission or scheduling workflow. The receiving party specifies the requirements to which the submitting party must attest. These requirements may comprise a baseline declaration - that the content was prepared in accordance with the applicable guidelines and contains no malicious code - and may be extended, at the receiving party's discretion, to reference specific requirements of this standard, such as conformance to an approved template, restriction to declared endpoints, or the absence of obfuscated or environment-conditional code, as supplier conformance to the standard is established.

5.2.6 CF01-05 - Pre-approval creative review workflow

Type	Applies To	Responsible Party	Enabling Party
Minimum	Content package	Media owner	CMS vendor

5.2.6.1 Requirement

Every creative package is reviewed against its declared package type before scheduling or publication. Remote domains, scripts, tags, images, styles, fonts and redirects are documented, expected and approved, and the package displays as expected during visual review.

5.2.6.2 Implementation guidance

Define a pre-approval workflow for every creative package before it is scheduled or published. Review each package against its declared package type: fully self-contained, semi self-contained or remote package. Confirm that the package content and observed behaviour match the declared type. Require remote domains, scripts, tags, images, styles, fonts and redirects to be documented before review. Approve only those that are expected and necessary for the package type and campaign. Check for obfuscated, suspicious or environment-conditional code that could change how the creative renders, load unapproved content or redirect users to unapproved destinations. Visually review each package as part of the approval step to confirm that it displays as expected.

5.2.7 CF01-06 - Capability manifests for executable content packages

Type	Applies To	Responsible Party	Enabling Party
Minimum	Content package	DCO agency, Verification provider	CMS vendor

5.2.7.1 Requirement

Each creative carries a declaration of the capabilities it requires, including network calls, data sources, and rendering modes.

5.2.7.2 Implementation guidance

Define a capability manifest for executable content packages and require it as part of submission or scheduling. The manifest should declare the capabilities the creative requires, including network calls, data sources, remote endpoints, rendering modes and any browser capabilities needed by the player. Use the manifest during package review to confirm that declared capabilities match the package type and observed behaviour. Use the approved manifest to drive player-side enforcement where supported. This may include sandboxed iframe profiles, Content Security Policy restrictions and controls that limit creatives to pre-approved data and content endpoints.

5.2.8 CF01-07 - Sandboxed pre-execution rendering and inspection

Type	Applies To	Responsible Party	Enabling Party
Recommended	Content package; Player rendering environment	Media owner	CMS vendor

5.2.8.1 Requirement

A content package is rendered in an isolated sandbox before approval. Content runtime behaviour is compared against expected behaviour and the package is rejected if there is a discrepancy.

5.2.8.2 Implementation guidance

Implement a pre-approval inspection gate for creative content before it is approved for scheduling or delivery to players. Render each submitted content package in an isolated sandbox that matches the production player rendering environment as closely as practicable, including the player version, browser engine, supported APIs, network policy and package configuration. Capture the creative's runtime behaviour, including access to remote assets and domains, script evaluation behaviour, execution duration and file access. To increase coverage, vary environment parameters such as date and time to capture behaviour that depends on those values. Compare observed sandbox behaviour with the expected behaviour identified during static code review. Reject the package if it loads unapproved external content, calls unexpected APIs, changes behaviour after a delay, attempts local file access, uses unexpected network destinations or otherwise differs from its declared behaviour.

5.2.9 CF01-08 - Inspectable, unobfuscated code as an onboarding condition

Type	Applies To	Responsible Party	Enabling Party
Recommended	Content package	Media owner	DCO agency, Verification provider

5.2.9.1 Requirement

Submitted package and tag code must be human-readable and understandable to a competent reviewer. Obfuscated code is rejected; minified code is acceptable only where source is available on request under NDA.

5.2.9.2 Implementation guidance

To make the review process repeatable, require DCOs, verification providers and other creative suppliers to provide unobfuscated code. Obfuscated code is rejected at onboarding. Minified code is acceptable only where source is available on request under NDA.

5.2.10CF01-09 - Versioned template requirement for submitted packages

Type	Applies To	Responsible Party	Enabling Party
Recommended	Content package	Media owner	DCO agency, Verification provider

5.2.10.1 Requirement

Reusable, templated package components - the consistent structure a vendor applies across many creatives - must be separately identified, versioned and approved once. Each package submission must declare the approved template version it is built on and expose only its variable parts for media owner review. This also supports delivery-chain traceability by making reusable package components and their changes identifiable across submissions and hand-offs.

5.2.10.2 Implementation guidance

For each vendor that builds, wraps or transforms packages (DCO agencies, verification providers, and any SSP or DSP that repackages), separate the reusable component from the variable parts that differ each time. Version and approve each reusable component once, before first production use. Thereafter each submission declares the approved template version it is built on and presents only its variable parts, so the media owner reviews only the delta. Any change to the reusable component requires a new version, submitted and validated before production use; changes confined to the declared variable fields do not. Include a version identifier in the package name and a manifest stating the template version, the variable fields populated, and what changed from the prior version.

5.2.11 CF01-10 - Production-matched sandbox review environment

Type	Applies To	Responsible Party	Enabling Party
Target	Player rendering environment	Media owner	-

5.2.11.1 Requirement

A UAT or sandbox environment is configured to match production exactly - screen dimensions, player metadata structure, network configuration - so environment-detection logic cannot identify the review context.

5.2.11.2 Implementation guidance

Configure the UAT or review environment to be indistinguishable from production - identical screen dimensions, the same player-metadata structure and variables, and equivalent network configuration and reachable endpoints - so environment-aware payloads cannot detect the review context and suppress behaviour. Re-confirm parity whenever production changes.

5.2.12 CF01-11 - Extended-duration pre-approval content testing

Type	Applies To	Responsible Party	Enabling Party
Target	Content package	Media owner	-

5.2.12.1 Requirement

Content packages are tested for an extended period before approval to ensure that malicious behaviour, such as time-delay attacks, can be detected before content packages are scheduled to players.

5.2.12.2 Implementation guidance

No further guidance beyond standard testing practice.

5.3 CF02 - Execution Integrity and Runtime Bounding

5.3.1 Overview

This control family ensures only approved content runs, and that it runs within defined runtime limits. It uses play-time integrity checks, CSP, sandboxed iframes, endpoint allow-listing and SRI to restrict what content can load, call and execute. It also constrains tracking pixels and other non-visible assets, embeds images and fonts where required, and prohibits remote executable content at the target tier.

5.3.2 CF02-01 - Package integrity at playtime

Type	Applies To	Responsible Party	Enabling Party
Recommended	Content package; Player rendering environment	Media owner	CMS vendor

5.3.2.1 Requirement

The player computes the hash of each creative before execution and compares it against the approved hash registered at scheduling. It refuses execution and plays fallback on mismatch.

5.3.2.2 Implementation guidance

Implement playtime integrity verification in the player before a content package is executed. When a package is approved for scheduling, register the approved cryptographic hash for each creative artefact that the player will execute or render. At playtime, compute the hash of the local package or file immediately before execution and compare it with the approved hash from the CMS or scheduling system. If any hash does not match, refuse execution of the package, record the integrity failure and play an approved fallback instead. Do not attempt to repair or continue using the modified local file. For programmatic or runtime-assembled content, apply the check to the stable wrapper or package artefact that is scheduled. Treat runtime-fetched verifier code or other external components as outside the hashed package unless they are separately controlled by the relevant remote content controls.

5.3.3 CF02-02 - Browser capability restrictions for player-rendered content

Type	Applies To	Responsible Party	Enabling Party
Recommended	Player rendering environment; Data feed; Content package	Media owner	CMS vendor

5.3.3.1 Requirement

The player rendering environment uses a Content Security Policy (CSP) to restrict browser capability.

5.3.3.2 Implementation guidance

Apply runtime restrictions using Content-Security-Policy (CSP) to every browser-rendered content package so creative content cannot use unnecessary browser capabilities.

Define and use the following CSP directives, as applicable:

- do not allow 'unsafe-eval', 'wasm-unsafe-eval' or 'unsafe-inline';
- set object-src to 'none' to prevent plugin content;
- set base-uri to 'none' or 'self' to prevent injected base URL changes;
- set form-action to 'none'; and
- restrict worker-src and frame-ancestors.

Prefer enforcing the policy with HTTP response headers on the player-rendered creative page. If HTTP headers cannot be added, use a CSP meta element as a fallback. Note: frame-ancestors is not supported when using meta elements.

Level 1 (Recommended): implement a common wrapper that removes browser capabilities not required by any approved creative type. This can also be implemented differently for each content package type - for example, fully self-contained packages might allow for stricter sandboxing policy.

Level 2 (Target): implement per-creative CSP profiles, based on the approved template or package declaration, so each creative receives only the browser capabilities it needs.

5.3.4 CF02-03 - Endpoint allow-listing for runtime content and data calls

Type	Applies To	Responsible Party	Enabling Party
Recommended	Content package; Data feed; Third-party data platform; Player rendering environment	DCO agency, Verification provider	-

5.3.4.1 Requirement

Creatives can only call pre-approved data and content endpoints, enforced at the player or at the SSP. Endpoints are pre-registered and reviewed at onboarding.

5.3.4.2 Implementation guidance

Apply runtime restrictions to every browser-rendered content package so creative content cannot load resources from unapproved origins. For example, this can be implemented by enforcing a Content-Security-Policy (CSP) that restricts resource loading to approved origins using explicit directives such as script-src, connect-src, frame-src, img-src, media-src and font-src. Prefer enforcing the policy with HTTP response headers on the player-rendered creative page. If HTTP headers cannot be added, use a CSP meta element as a fallback. Make sure the policy set for a creative aligns with the package type and approved remote domains / destinations.

5.3.5 CF02-04 - Remote image and font restriction

Type	Applies To	Responsible Party	Enabling Party
Recommended	Content package	DCO agency, Verification provider	-

5.3.5.1 Requirement

Submitted packages do not contain remotely loaded images or fonts.

5.3.5.2 Implementation guidance

Apply this requirement to every party involved in creating, modifying or extending a package, including verification providers, DCOs, creative providers and other relevant parties. Embed images and fonts directly in the HTML package using Data URIs instead of referencing external resources. Reject packages that do not conform.

5.3.6 CF02-05 - Subresource Integrity (SRI) enforcement for dynamic assets

Type	Applies To	Responsible Party	Enabling Party
Recommended	Content package	DCO agency, Verification provider	-

5.3.6.1 Requirement

Submitted packages use Subresource Integrity (SRI) for every dynamically loaded script and style.

5.3.6.2 Implementation guidance

Apply this requirement to every party that creates, modifies or extends a package, including verification providers, DCOs, creative providers and other relevant parties. Ensure all eligible dynamically loaded content, such as scripts and styles, is protected with SRI (Subresource Integrity). When adding new resources, pin them to an approved version, generate the expected integrity hash, and update the package manifest or HTML references before release.

5.3.7 CF02-06 - Restrictive CSP for remote content

Type	Applies To	Responsible Party	Enabling Party
Recommended	Content package	DCO agency, Verification provider	-

5.3.7.1 Requirement

Remotely loaded content uses a restrictive Content-Security-Policy that blocks dynamic code execution and unnecessary active content.

5.3.7.2 Implementation guidance

Apply this requirement to every party that creates, modifies or extends a package, including verification providers, DCOs, creative providers and other relevant parties. Implement a restrictive Content-Security-Policy for all remotely loaded content. Block dynamic code execution and unnecessary active content. Do not allow 'unsafe-eval', 'wasm-unsafe-eval' or 'unsafe-inline' unless required. Set object-src to 'none' to block plugin content. Set form-action to 'none' unless the content has a documented requirement to submit forms. Restrict script-src, connect-src, worker-src and frame-ancestors to the minimum required domains.

5.3.8 CF02-07 - Constraint of non-visible image tags in creatives

Type	Applies To	Responsible Party	Enabling Party
Recommended	Content package	DCO agency, Verification provider	-

5.3.8.1 Requirement

Image tags in creatives that are intended to be non-visible, such as tracking pixels, are rendered off-screen or constrained to a fixed small size.

5.3.8.2 Implementation guidance

When accepting or building content packages, explicitly classify image tags that are intended to be non-visible, such as tracking pixels. Render each non-visible image outside the visible canvas or constrain it with fixed width and height values that prevent the remote image from affecting the creative layout. Do not rely on the remote image's natural dimensions or response content to keep the image hidden. Validate this during content package review or automated creative linting by checking that each non-visible img tag has approved positioning or fixed size attributes or styles.

5.3.9 CF02-08 - Browser capability sandboxing for executable HTML content

Type	Applies To	Responsible Party	Enabling Party
Recommended	Player rendering environment; Content package	Media owner	CMS vendor

5.3.9.1 Requirement

The player rendering environment uses a sandboxed iframe to restrict browser capabilities for executable HTML content.

5.3.9.2 Implementation guidance

Implement iframe sandboxing in the player rendering environment for executable HTML content. For example, the player should load the primary creative HTML file inside a wrapper page that embeds it in a sandboxed iframe. Start with an empty sandbox attribute and add only the capabilities needed for the package to render, such as allow-scripts or allow-same-origin. Do not grant allow-popups, allow-top-navigation or allow-forms unless they are explicitly required for the approved package behaviour.

Level 1 (Recommended): implement a common wrapper that removes browser capabilities not required by any approved creative type. This can also be implemented differently for each content package type - for example, fully self-contained packages might allow for stricter sandboxing policy.

Level 2 (Target): implement per-creative sandbox profiles, based on the approved template or package declaration, so each creative receives only the browser capabilities it needs.

5.3.10 CF02-09 - Prohibition of remote executable content loading

Type	Applies To	Responsible Party	Enabling Party
Target	Content package	DCO agency, Verification provider	-

5.3.10.1 Requirement

Submitted packages do not load executable or visible content remotely, such as JavaScript or iframes.

5.3.10.2 Implementation guidance

Apply this requirement to every party that creates, modifies or extends a package, including verification providers, DCOs, creative providers and other relevant parties. Ensure submitted packages do not load executable or visible content from remote locations, including JavaScript, iframes, fonts, images or other runtime assets.

5.4 CF03 - Runtime Monitoring and Detection

5.4.1 Overview

This control family detects live campaign behaviour that diverges from the approved baseline. It monitors data-feed responses and runtime behaviour so unexpected content, schema changes, code paths or network calls can trigger fallback or investigation.

5.4.2 CF03-01 - Data response anomaly detection with approved fallback

Type	Applies To	Responsible Party	Enabling Party
Target	Data feed; Third-party data platform	Media owner	DCO agency, Verification provider

5.4.2.1 Requirement

Dynamic content packages that pull from external data feeds, such as weather and temperature data, live news content, pricing feeds, or other runtime data sources, must have their feed responses monitored independently of the creative itself.

5.4.2.2 Implementation guidance

Monitor data-feed responses used by dynamic content packages before they are rendered. Validate that each response conforms to the expected schema, falls within acceptable value ranges, and does not contain inappropriate, malicious or unexpected content. If a response is unexpected, trigger fallback to a pre-approved static creative rather than allowing the creative to render unpredictable content.

5.4.3 CF03-02 - Behavioural baseline monitoring of live campaigns

Type	Applies To	Responsible Party	Enabling Party
Target	Content package	Media owner	-

5.4.3.1 Requirement

Execution behaviour during live campaigns is monitored against the baseline established at UAT and early campaign. New code execution paths or external network calls absent from the baseline trigger an alert.

5.4.3.2 Implementation guidance

Establish a behavioural baseline for each live campaign using the behaviour observed during UAT and the early campaign period. Monitor execution during delivery for new code paths, remote domains, network calls or other runtime behaviour not present in that baseline. Treat unexpected divergence as an alert requiring investigation and, where the divergence could affect displayed content or player security, suspend the affected creative or move to an approved fallback while the issue is assessed.

5.5 CF04 - Origin to Destination Authenticity

5.5.1 Overview

This control family verifies that content and vendor code come from approved sources and that screens display the approved schedule. It combines package authenticity checks with secondary screen monitoring to detect substitution, defacement, blank screens or inappropriate content at display time.

5.5.2 CF04-01 - Secondary screen-content monitoring

Type	Applies To	Responsible Party	Enabling Party
Target	Player device	Media owner	-

5.5.2.1 Requirement

Secondary screen monitoring technology is used to detect malicious content being displayed on players.

5.5.2.2 Implementation guidance

Deploy secondary screen monitoring for player devices so the organisation can verify what is actually being displayed, not only what the content schedule says should be displayed. This can be implemented using periodic screenshot capture from the player, display-output capture, or an independent camera or screen-sensing capability where privacy and site conditions allow. Compare captured screen content against the approved playlist, campaign schedule or known-good visual fingerprints. Use image matching, perceptual hashing, OCR and content moderation detection to identify unexpected content, blank screens, defacement, error messages, overlays or content that does not match the expected schedule. Consider implementing this control for higher-risk screens that may cause higher reputational impact if inappropriate content is displayed.

5.5.3 CF04-02 - Authenticity verification for inter-organisation content packages

Type	Applies To	Responsible Party	Enabling Party
Target	Content package	Media owner	-

5.5.3.1 Requirement

Content packages transferred between organisations can be authenticated before acceptance, allowing recipients to verify the package source.

5.5.3.2 Implementation guidance

Digitally sign each content package before it is transferred between organisations. Package the content into a single archive, such as a ZIP file, then generate a detached signature for that archive. Verify the digital signature before accepting, storing, scheduling or distributing the content package. Reject packages with a missing signature, an invalid signature, or a signature from an untrusted key. For example, this could be implemented using GPG. Establish a one-time trust relationship between each participating organisation to exchange and verify GPG public keys. Each sending organisation signs the packaged content with its private key and sends both the content package and the signature file, such as a .sig file. Each receiving organisation verifies the signature against the sender's pre-exchanged public key before committing the package to organisational storage. Protect signing private keys in an approved secrets management or key management process. Limit signing access to authorised users or systems and rotate keys if compromise is suspected.

5.6 CF05 - Custody and Transfer Protection

5.6.1 Overview

This control family protects content in transit between organisations and systems. It requires authenticated, encrypted transfer channels so packages cannot be altered or accepted from an unverified party without detection.

5.6.2 CF05-01 - TLS certificate validation for content package transfers

Type	Applies To	Responsible Party	Enabling Party
Minimum	Inter-party transfer channel	Media owner	-

5.6.2.1 Requirement

TLS with certificate chain validation is used for all HTTP-based transfer of content packages.

5.6.2.2 Implementation guidance

Use HTTPS for HTTP-based content package transfers, and require valid TLS certificate chain validation. Do not disable certificate validation or accept untrusted certificates in transfer tooling, scripts, integrations or operational workarounds. Where a transfer fails certificate validation, treat the failure as a transfer integrity issue and investigate before accepting the package.

5.6.3 CF05-02 - Encrypted and authenticated file transfer channel

Type	Applies To	Responsible Party	Enabling Party
Recommended	Inter-party transfer channel	Media owner	-

5.6.3.1 Requirement

File transfers not covered by signature validation use an encrypted and authenticated communication channel, such as SFTP.

5.6.3.2 Implementation guidance

Use an encrypted and authenticated transfer protocol for file transfers that do not have separate content signature validation. Suitable options include SFTP with SSH host key validation, HTTPS with valid TLS certificate validation, or a managed file transfer service that provides equivalent channel authentication and encryption.

5.7 CF06 - Player Physical Integrity

5.7.1 Overview

This control family protects player devices in public or semi-public locations. It makes physical tampering visible, protects stored data if a device is removed, and adds alerting or fallback behaviour where interference is detected.

5.7.2 CF06-01 - Tamper-evident physical enclosure of player devices

Type	Applies To	Responsible Party	Enabling Party
Minimum	Player device	Media owner	-

5.7.2.1 Requirement

Player devices are installed in locked enclosures covering HDMI, USB, power and storage access points, with tamper-evident seals on every panel.

5.7.2.2 Implementation guidance

Install player devices in lockable, tamper-evident enclosures that prevent access to HDMI, USB, power, storage and service ports. Apply uniquely numbered seals to each removable or opening panel, record seal IDs, and inspect them during installation, maintenance and periodic site checks. Replace broken or missing seals only through an approved maintenance process, and document any discrepancy as a physical access incident.

5.7.3 CF06-02 - Full-disk encryption of player storage

Type	Applies To	Responsible Party	Enabling Party
Minimum	Player device; Player content storage	Media owner	-

5.7.3.1 Requirement

Full-disk encryption is used on every player as a standard build requirement to protect the operating system, configuration and stored content if a player is removed or accessed offline.

5.7.3.2 Implementation guidance

Build full-disk encryption into the standard player device image and deployment process so every player storage device is encrypted before installation. For Windows-based players, enable BitLocker with TPM protection and Secure Boot. Validate the status of full-disk encryption, BitLocker and Secure Boot during build acceptance and periodic compliance checks, such as through an endpoint management compliance policy.

5.7.4 CF06-03 - Physical tethering of player cabling and connectors

Type	Applies To	Responsible Party	Enabling Party
Recommended	Player device	Media owner	-

5.7.4.1 Requirement

HDMI, USB and power cables are physically tethered so disconnection requires tools and leaves visible damage.

5.7.4.2 Implementation guidance

Consider physical tethering for screens in higher-risk areas, where screens cannot be easily monitored or where inappropriate displayed content would cause higher reputational impact. Tether HDMI, USB and power connections so casual disconnection or replacement is difficult and leaves visible evidence. Include tethering checks in installation, maintenance and site inspection procedures.

5.7.5 CF06-04 - Tamper detection and alerting to the CMS

Type	Applies To	Responsible Party	Enabling Party
Recommended	Player device	Media owner	-

5.7.5.1 Requirement

A mechanical tamper switch on the enclosure, or HDMI signal-loss detection, triggers a screen blank or fallback image and a real-time alert to the CMS.

5.7.5.2 Implementation guidance

Connect enclosure tamper switches or HDMI signal-loss detection to the player or CMS monitoring workflow. Configure the player to blank the screen or display an approved fallback image when tampering is detected and raise a real-time alert to the CMS or operational monitoring queue. Treat tamper alerts as incidents requiring investigation before normal playback resumes.

5.8 CF07 - Player and Network Segregation

5.8.1 Overview

This control family limits what a player can reach and who can modify its content storage. It uses network isolation, egress control, management-port restrictions and file-integrity monitoring to contain compromised players and local site systems.

5.8.2 CF07-01 - Network isolation between players and management infrastructure

Type	Applies To	Responsible Party	Enabling Party
Minimum	Player network	Media owner	-

5.8.2.1 Requirement

The player network can only reach management and backend infrastructure on explicitly required ports, blocking pivot from a compromised player into backend systems.

5.8.2.2 Implementation guidance

Segment the player network from management and backend infrastructure using controls appropriate to the environment, such as VLANs, firewall rules, router ACLs, cloud security groups or zero trust network policies. Apply a default-deny policy from the player network to other networks, including management and backend networks. Permit only the specific destination systems and ports required for player operation, such as content retrieval, telemetry, health checks or time synchronisation.

5.8.3 CF07-02 - Isolation of player content storage from local networks

Type	Applies To	Responsible Party	Enabling Party
Minimum	Player content storage; Player network	Media owner	-

5.8.3.1 Requirement

The player content directory is not accessible via SMB or any other network file share from the local site network.

5.8.3.2 Implementation guidance

Where players use file-based content storage, do not expose the local content directory, creative repository or cache directories to the local site network through SMB, NFS, AFP, WebDAV or any other network file-sharing protocol. Content should be written to player storage only through the approved content distribution or management channel. This prevents local network users or compromised site systems from adding, replacing or copying creative content outside the approved scheduling and validation workflow.

5.8.4 CF07-03 - Controlled and monitored network egress for player traffic

Type	Applies To	Responsible Party	Enabling Party
Recommended	Player network	Media owner	-

5.8.4.1 Requirement

All player network egress is routed through a controlled proxy or gateway with traffic monitoring and anomalous-request detection.

5.8.4.2 Implementation guidance

Route outbound player network traffic through an approved egress control point, such as an explicit proxy, secure web gateway or network firewall. Block direct internet access from player devices except where a documented exception is required. Maintain an allow-list of approved destinations, protocols and ports for content retrieval, player management, software updates, telemetry, DNS, time synchronisation and other approved use cases. Enable logging and monitoring at the proxy or firewall, and alert on unusual destinations, unexpected protocols or ports, and traffic from players that should only display static content.

5.8.5 CF07-04 - Player management port restriction

Type	Applies To	Responsible Party	Enabling Party
Recommended	Player network	Media owner	-

5.8.5.1 Requirement

Player management network ports are only accessible from approved management stations.

5.8.5.2 Implementation guidance

Configure host firewalls and network firewalls so player management services and ports are reachable only from approved management stations or management subnets. Block management ports from all other sources. On Windows-based players, this commonly includes SMB on TCP 139 and 445, RDP on TCP 3389, and WinRM on TCP 5985 and 5986. On Linux-based players, this commonly includes SSH on TCP 22.

5.8.6 CF07-05 - Continuous file-integrity monitoring of player content storage

Type	Applies To	Responsible Party	Enabling Party
Target	Player content storage	Media owner	CMS vendor

5.8.6.1 Requirement

Real-time alerts are generated for any file modification in the player content directory that does not originate from an authenticated sync process.

5.8.6.2 Implementation guidance

Implement continuous file-integrity monitoring on the player content directory and forward alerts to the operational monitoring or SIEM workflow. On Windows players, this may use Sysmon with SIEM forwarding, or Windows Security Auditing with Object Access auditing and a SACL on the player content directory for write, create, delete and rename operations. Alert when changes occur from an unexpected process or outside the approved content synchronisation channel.

5.9 CF08 - Identity, Access and Accountability

5.9.1 Overview

This control family governs access to delivery-chain platforms and accountability for actions on them. It uses MFA, least privilege, unique credentials, vaulting, audit logging and separation of duties to reduce the impact of stolen credentials or excessive permissions.

5.9.2 CF08-01 - Multi-factor authentication on content and scheduling platforms

Type	Applies To	Responsible Party	Enabling Party
Minimum	CMS / scheduling platform; SSP platform; DSP platform; Verifier platform; DCO platform; Third-party data platform	Media owner, CMS vendor, SSP, DSP, verification provider	-

5.9.2.1 Requirement

MFA is enforced on every user account capable of scheduling, content submission or retrieval, or platform administration.

5.9.2.2 Implementation guidance

Where SSO is used for content scheduling, content submission, content retrieval or platform administration, enforce MFA through the identity provider for every account with access to those functions. Where SSO is not available, enforce MFA using the platform's native authentication controls. Apply this to every relevant platform in the pathway, including DSP and SSP platforms where programmatic advertising is used. Where a platform cannot support MFA or SSO, apply compensating access controls such as approved source IP restrictions or access through a managed jump host.

5.9.3 CF08-02 - Unique credentials per device and per account

Type	Applies To	Responsible Party	Enabling Party
Minimum	Player device; CMS / scheduling platform	Media owner	CMS vendor

5.9.3.1 Requirement

Locally configured accounts use unique credentials per device. Shared credentials are not used across sites, teams or devices.

5.9.3.2 Implementation guidance

Use unique, randomly generated credentials for all local accounts and store them in a centrally managed password vault. Do not share passwords across devices. For Windows-based devices, deploy Windows LAPS to manage and automatically rotate the local Administrator password across the fleet. For Linux-based devices, use SSH keys or an organisation-approved endpoint management or privileged access management tool to issue unique local credentials per device.

5.9.4 CF08-03 - Attributable audit logging of platform actions

Type	Applies To	Responsible Party	Enabling Party
Minimum	CMS / scheduling platform	Media owner	CMS vendor

5.9.4.1 Requirement

Every scheduling, content upload, login and platform action is logged with identity, timestamp, source and action detail, retained for a defined period and query-able for incident response.

5.9.4.2 Implementation guidance

Capture attributable audit logs for all actions that can affect content, schedules, accounts, permissions and platform configuration. Retain logs for a defined period and ensure they can be searched during incident response. Logs should identify the user or service account, source, timestamp, action performed and affected object.

5.9.5 CF08-04 - Machine-generated password vaulting requirement

Type	Applies To	Responsible Party	Enabling Party
Minimum	Player device; CMS / scheduling platform; SSP platform; DSP platform; Verifier platform; DCO platform	Media owner, CMS vendor, DSP, SSP, Verification provider, DCO agency	-

5.9.5.1 Requirement

Where SSO is not used, complex, machine-generated passwords are required.

5.9.5.2 Implementation guidance

Store passwords in a centrally managed password vault solution. Do not store platform, local or shared credentials in spreadsheets, documents, tickets, email, chat messages or unmanaged browser password stores.

5.9.6 CF08-05 - Least-privilege filesystem permissions for player service accounts

Type	Applies To	Responsible Party	Enabling Party
Recommended	Player content storage; Player software	Media owner	CMS vendor

5.9.6.1 Requirement

Player software runs under a dedicated service account with least-privilege filesystem access. Other local accounts cannot read or modify player content or software unless explicitly required.

5.9.6.2 Implementation guidance

Run the player software under a dedicated, non-administrator service account. The account must not be a local administrator on Windows or have root privileges on Linux-based players. Apply filesystem permissions to the player software, media cache, package staging, configuration and log directories so the player service account has only the read, write or execute permissions it requires, and other local accounts cannot access content unless explicitly required.

5.9.7 CF08-06 - Single sign-on enforcement on delivery chain platforms

Type	Applies To	Responsible Party	Enabling Party
Recommended	CMS / scheduling platform; SSP platform; DSP platform; Verifier platform; DCO platform; Third-party data platform	Media owner, CMS vendor, SSP, DSP, verification provider	-

5.9.7.1 Requirement

Every scheduling and content submission platform authenticates users using SSO. No local accounts other than break-glass accounts are used.

5.9.7.2 Implementation guidance

Use SSO for each content scheduling, content submission and platform administration system where the platform supports it. Where a platform cannot support SSO, apply compensating access controls such as approved source IP restrictions or access through a managed jump host.

5.9.8 CF08-07 - Centralised credential vaulting and rotation

Type	Applies To	Responsible Party	Enabling Party
Recommended	CMS / scheduling platform; SSP platform; DSP platform; Verifier platform; DCO platform	Media owner, CMS vendor, SSP, DSP, verification provider	-

5.9.8.1 Requirement

A centralised, access-controlled vault stores all unique credentials with a mandatory rotation schedule, making credential uniqueness operationally viable at fleet scale.

5.9.8.2 Implementation guidance

Store all shared, local and platform credentials in an organisation-approved password vault. Grant vault access to named users using role-based access control and limit each user to the credentials required for their role. Protect vault access with MFA. Where supported, require check-out workflows, time-limited access and automatic rotation after use for privileged credentials. Retain audit logs for vault sign-ins, credential views, check-outs, password reveals, password changes and administrative actions.

5.9.9 CF08-08 - Least-privilege access and audit logging for creative repositories

Type	Applies To	Responsible Party	Enabling Party
Recommended	CMS / scheduling platform; SSP platform; DSP platform; Verifier platform; DCO platform; Player devices	Media owner, CMS vendor, SSP, DSP, verification provider	-

5.9.9.1 Requirement

Creative file repositories on players enforce least-privilege access using named users or role-based groups, restrict modification and deletion of approved files, and retain audit logs for repository changes.

5.9.9.2 Implementation guidance

Grant access to repositories that store creative files, source assets and intermediate files using least privilege. Use named users or role-based groups rather than shared accounts. Grant read, write, modify and delete permissions only where each role requires them. Limit the ability to modify or delete approved files, and retain audit logs for file creation, modification, deletion and permission changes. Review access periodically and remove access that is no longer required.

5.9.10 CF08-09 - Behavioural anomaly detection on platform activity

Type	Applies To	Responsible Party	Enabling Party
Recommended	CMS / scheduling platform; SSP platform	Media owner	CMS vendor, SSP vendor

5.9.10.1 Requirement

Real-time alerts are generated for unusual geography or hours, bulk changes, scheduling outside normal patterns, departed-staff logins and anomalous programmatic win rates.

5.9.10.2 Implementation guidance

Monitor platform activity for behaviour that differs from normal operational patterns. Alerts should cover unusual locations, unusual hours, bulk scheduling or content changes, logins from accounts that should no longer be active, and programmatic patterns that may indicate account compromise or manipulation.

5.9.11 CF08-10 - Separation of duties between content submission and activation

Type	Applies To	Responsible Party	Enabling Party
Target	CMS / scheduling platform	Media owner	CMS vendor

5.9.11.1 Requirement

No single account can both upload and schedule content without secondary authorised approval. A two-person rule is required for content activation so one person cannot upload and then schedule content at once.

5.9.11.2 Implementation guidance

Configure the submission and activation workflow so the account that uploads or submits content cannot independently activate that content for display. Require secondary authorised approval for activation, particularly for bulk scheduling changes or changes that affect many screens.

5.9.12 CF08-11 - Phishing-resistant multi-factor authentication on content and scheduling platforms

Type	Applies To	Responsible Party	Enabling Party
Target	Media owner, CMS vendor, SSP, DSP, verification provider	Media owner, CMS vendor, DSP, SSP, Verification provider, DCO agency	-

5.9.12.1 Requirement

Phishing-resistant MFA is enforced on every user account capable of scheduling, content submission or platform administration.

5.9.12.2 Implementation guidance

Protect administrator and operator access to the CMS, scheduling platform, SSP, DSP and remote player access with phishing-resistant MFA. Acceptable controls include FIDO2 or WebAuthn hardware security keys, platform passkeys bound to managed devices, or certificate-based authentication using managed devices. Do not rely on push notifications, SMS, voice calls or email OTP for these systems. Where conditional access is available, require access from managed devices and block legacy authentication methods that cannot enforce MFA.

5.10 CF09 - Supply-chain Assurance and Blast-radius Control

5.10.1 Overview

This control family sets security obligations for suppliers and limits the blast radius of supplier compromise. It uses risk assessment, contractual obligations, notification, tenant isolation, signed updates, staged rollout and testing to reduce industry-wide impact.

5.10.2CF09-01 - Third-party risk assessment of creative-chain partners

Type	Applies To	Responsible Party	Enabling Party
Minimum	N/A	Media owner	DCO agency, SSP, DSP, CMS vendor

5.10.2.1 Requirement

Third-party risk assessments are performed for all partners involved in creative preparation, approval, submission and scheduling.

5.10.2.2 Implementation guidance

Perform a third-party risk assessment for each DCO, SSP, DSP and CMS partner that can affect content selection, approval, delivery or display. Assess the partner's cyber security controls relevant to the service provided, including identity and access management, MFA, privileged access, change control, logging and monitoring, operating-system hardening, incident notification, data handling, vulnerability management and subcontractor management. Where creative approval or content selection is delegated to a third party, confirm how content is reviewed, who can approve or change it, how unvetted dynamic content is controlled, and whether the residual risk is within appetite.

5.10.3 CF09-02 - Contractual security attestation with notification obligations

Type	Applies To	Responsible Party	Enabling Party
Minimum	N/A	Media owner	DCO agency, SSP, DSP, CMS vendor

5.10.3.1 Requirement

Contractual obligations are established with every supply chain party for annual security attestation against defined DOOH-specific risks, framework and guidelines.

5.10.3.2 Implementation guidance

Include annual security attestation obligations in contracts with each relevant supply-chain party. The attestation should reference the agreed DOOH-specific risk profile and applicable control baseline, and should be updated when the supplier service, ownership or architecture materially changes.

5.10.4 CF09-03 - Incident notification within a defined window

Type	Applies To	Responsible Party	Enabling Party
Minimum	N/A	Media owner	DCO agency, SSP, DSP, CMS vendor

5.10.4.1 Requirement

Contracts require notification to the media owner within a defined window, typically 24 to 72 hours, of any suspected compromise of a platform, build system or data provider used in active campaigns.

5.10.4.2 Implementation guidance

Define the notification window, recipient, minimum information to be provided and escalation path in supplier contracts and operational procedures. Notifications should cover suspected compromise of platforms, build systems, data providers, content delivery paths and any system that can affect active campaigns.

5.10.5 CF09-04 - Security-specific contractual requirements

Type	Applies To	Responsible Party	Enabling Party
Minimum	N/A	Media owner	CMS vendor

5.10.5.1 Requirement

Contracts with each supply chain vendor identify the applicable security requirements derived from the agreed control baseline and require those requirements to be included in the contract or explicitly agreed by the relevant parties before continued access is permitted.

5.10.5.2 Implementation guidance

Translate the agreed control baseline into contract requirements for each supplier role. Where a requirement cannot be met immediately, record the exception, compensating control, owner and agreed timeframe before continued access is permitted.

5.10.6 CF09-05 - Signed software updates with player-side verification

Type	Applies To	Responsible Party	Enabling Party
Recommended	CMS player software release; Player software	CMS vendor	-

5.10.6.1 Requirement

Every CMS player software update is signed by the vendor. The player software verifies the signature and refuses unsigned or invalidly signed updates before installation.

5.10.6.2 Implementation guidance

Sign each player software update as part of the release process and verify the signature on the player before installation. Players should refuse unsigned updates, invalid signatures and signatures from untrusted keys. Protect signing keys through an approved key management process and rotate keys if compromise is suspected.

5.10.7CF09-06 - Annual adversarial testing of shared platforms

Type	Applies To	Responsible Party	Enabling Party
Recommended	All	DCO agency, Verification provider, SSP, Third-party data provider	-

5.10.7.1 Requirement

Annual penetration testing is conducted for all relevant supply chain participants, covering DOOH-specific paths that SOC 2 and external ratings do not probe.

5.10.7.2 Implementation guidance

Conduct annual adversarial testing of shared platforms and supplier systems that can affect content selection, approval, delivery, runtime behaviour or display. The scope should include DOOH-specific paths and should not rely solely on generic assurance reports or external security ratings. Refer to the technical security assessment scopes in this Guideline for example scope areas.

5.10.8 CF09-07 - Staged rollout of player software updates

Type	Applies To	Responsible Party	Enabling Party
Recommended	CMS player software release; Player software	CMS vendor	-

5.10.8.1 Requirement

Updates are deployed to a limited canary cohort with a defined observation window before fleet-wide release, limiting blast radius in case of malicious updates.

5.10.8.2 Implementation guidance

Release player software updates in stages, beginning with a limited canary cohort. Monitor the canary cohort for health, playback, integrity and security indicators during a defined observation window before broader deployment. Halt or roll back the rollout if unexpected behaviour is detected.

5.10.9 CF09-08 - Tenant isolation within shared platforms and storage

Type	Applies To	Responsible Party	Enabling Party
Target	CMS / scheduling platform	CMS vendor	-

5.10.9.1 Requirement

Shared CMS and SSP infrastructure provides logical and technical separation of each tenant's scheduling data, content, credentials and storage at the platform layer.

5.10.9.2 Implementation guidance

Enforce tenant isolation across scheduling data, content, credentials, configuration and storage in shared platforms. Validate isolation through design review and security testing, including attempts to access or affect another tenant's data, schedules, content or credentials.

5.11CF10 - Baseline System and Operational Hygiene

5.11.1 Overview

This control family covers baseline security hygiene for DOOH players, platforms, networks and vendor pipelines. It expects recognised practices for patching, hardening, monitoring, incident response, secure development and recovery so the DOOH-specific controls remain effective.

5.11.2 CF10-01 - Patch and vulnerability management

Type	Applies To	Responsible Party	Enabling Party
Minimum	All	All	-

5.11.2.1 Requirement

Known vulnerabilities in systems that display and support DOOH content are identified and remediated within risk-based timeframes. This includes player operating systems and firmware, rendering engines and browsers, CMS platforms and other supply chain supporting platforms.

5.11.2.2 Implementation guidance

Apply existing industry best practice for patch and vulnerability management. The process should cover all in-scope DOOH assets, whether or not dynamic content is running, because unpatched players or platforms can undermine the player-side and platform controls established elsewhere in this Guideline.

5.11.3 CF10-02 - Change management

Type	Applies To	Responsible Party	Enabling Party
Minimum	All	All	-

5.11.3.1 Requirement

Changes to the configuration of DOOH players and delivery-chain platforms, and to vendor release pipelines, are controlled, reviewed and recorded.

5.11.3.2 Implementation guidance

Apply existing industry best practice for change management. The process should cover configuration and code for all in-scope DOOH systems, because uncontrolled change can silently reintroduce risk or break an approved baseline.

5.11.4 CF10-03 - System hardening and endpoint protection

Type	Applies To	Responsible Party	Enabling Party
Minimum	All	All	-

5.11.4.1 Requirement

DOOH players and the management workstations used to operate them are hardened to a secure configuration baseline and protected against malware.

5.11.4.2 Implementation guidance

Apply existing industry best practice for system hardening and endpoint protection. Hardening should reduce unnecessary services, accounts, default credentials and exposed interfaces, and endpoint protection should be maintained on in-scope player and management endpoints where supported.

5.11.5 CF10-04 - Asset management

Type	Applies To	Responsible Party	Enabling Party
Minimum	All	All	-

5.11.5.1 Requirement

An accurate and current inventory is maintained of the DOOH assets and software that store, process or deliver content, including the player fleet, delivery-chain platforms, and the software and third-party components running on them.

5.11.5.2 Implementation guidance

Apply existing industry best practice for asset and software inventory. The inventory should support patching, monitoring, content takedown and recovery, because an unrecorded player or unmanaged software component is a blind spot that the controls in this Guideline cannot reach.

5.11.6 CF10-05 - Malware defences

Type	Applies To	Responsible Party	Enabling Party
Minimum	All	All	-

5.11.6.1 Requirement

Anti-malware and endpoint protection are deployed and maintained on DOOH assets and the systems that manage them, including players, management workstations and delivery-chain platforms.

5.11.6.2 Implementation guidance

Apply existing industry best practice for malware defences. Coverage should include all in-scope DOOH assets capable of running endpoint protection, because malware on a player or management workstation can subvert player-side and access controls.

5.11.7 CF10-06 - Incident response

Type	Applies To	Responsible Party	Enabling Party
Minimum	All	All	-

5.11.7.1 Requirement

Each organisation maintains an incident response capability for incidents that could affect DOOH content, platforms, players, vendor pipelines or supply-chain services.

5.11.7.2 Implementation guidance

Apply existing industry best practice for incident response. Response procedures should include escalation paths, supplier notification, content suspension or takedown, evidence preservation and recovery of affected players or platforms where relevant.

5.11.8 CF10-07 - Secure development of supply-chain code

Type	Applies To	Responsible Party	Enabling Party
Recommended	Vendor-shipped code and platforms	All	-

5.11.8.1 Requirement

Code that vendors ship into the DOOH delivery chain, including CMS player software, verifier code and DCO engine code, is produced under a secure development lifecycle.

5.11.8.2 Implementation guidance

Apply existing industry best practice for secure development, such as the NIST Secure Software Development Framework or equivalent. This applies to every party that ships executable code into the chain, because vendor-shipped code can execute across many campaigns and screens at once.

5.11.9 CF10-08 - Backup and recovery

Type	Applies To	Responsible Party	Enabling Party
Recommended	All	All	-

5.11.9.1 Requirement

DOOH player images, configuration and CMS scheduling data can be restored from a verified, known-good source.

5.11.9.2 Implementation guidance

Apply existing industry best practice for backup and recovery. Backups should support recovery of the in-scope systems that hold or schedule DOOH content.

5.11.10 CF10-09 - Security monitoring and logging

Type	Applies To	Responsible Party	Enabling Party
Recommended	All	All	-

5.11.10.1 Requirement

Security-relevant events across all in-scope infrastructure, including player devices, networks and delivery-chain platforms, are logged, retained and monitored so anomalous or malicious activity can be detected and investigated.

5.11.10.2 Implementation guidance

Apply existing industry best practice for security monitoring and logging. Logs should support investigation of platform access, administrative action, player activity, network egress, update activity and content-related changes across the systems that create, store, transfer or play creatives.

6 Appendix — Asset Definitions

Each sub-control in this Guideline references one or more assets from the register below. Assets are grouped by where they sit in the DOOH ecosystem: the player environment, shared platforms, content and packages in transit and vendor-shipped code. Because supply chains vary, not every asset will exist in every participant's environment. The asset references on each sub-control indicate where the control applies, so participants can quickly identify which controls are relevant to their specific configuration.

6.1 Player Environment

Player environment assets are the physical and software components that a media owner operates to schedule, store, render, and display content on screen. These assets sit entirely within the media owner's operational boundary. The media owner is accountable for all of them, with the CMS vendor sharing responsibility where it supplies or operates the player software. Controls referencing these assets apply primarily to media owners and CMS or player software vendors.

Asset	Definition	Accountable / Interested Parties
Player device	Physical playback unit: hardware, OS, firmware, ports, and cabling.	Media owner
Player software	Player application running on the device, including the update channel and local configuration.	Media owner (operates); CMS vendor (supplies)
Player content storage	Filesystem holding cached creatives and packages awaiting playback.	Media owner
Player rendering environment	Execution sandbox or browser running creative code at play time.	Media owner (configures); CMS vendor (capability)
Player network	Network segments including management connectivity and outbound egress.	Media owner

6.2 Platforms

Platform assets are the software systems through which content is scheduled, traded, verified, and delivered. They are operated by a range of participants including media owners, CMS vendors, SSPs, DSPs, verification providers, DCO agencies, and third-party data providers. Several of these platforms serve multiple participants simultaneously, which means a compromise of any one platform can have a broad blast radius across the industry. Controls referencing these assets apply to the party that operates each platform and, where relevant, to the media owners and advertisers that procure and depend on them.

Asset	Definition	Accountable / Interested Parties
CMS / scheduling platform	Platform used to ingest, schedule, moderate, and dispatch content to players.	CMS vendor (hosted) or media owner (self-hosted)
SSP platform	Exposes media owner inventory to programmatic demand and dispatches play instructions.	SSP
DSP platform	Buys impressions and submits creative or wrapper code into the delivery chain.	DSP

Verifier platform	Serves verification wrapper, receives telemetry, and produces verification reporting.	Verification provider
DCO platform	Builder environment where dynamic templates and feed bindings are configured.	DCO agency
Third-party data platform	Backend system serving runtime data responses to dynamic creatives, such as content delivery networks (CDNs) and endpoints.	Third-party data provider

6.3 Content and Packages in Transit

These assets represent creative content and the channels through which it moves between parties. Unlike platform or player assets, ownership and accountability for in-transit assets shifts depending on the delivery pathway and who holds custody at each point. A content package in direct delivery is owned by the creative agency until it is handed to the media owner; in the programmatic pathway, the same package may pass through a DSP, SSP, and verifier before it reaches the player. Controls referencing these assets apply to whichever party holds custody at the relevant point in the chain, and participants should use the pathway descriptions in the Framework to determine where their obligations apply.

Asset	Definition	Accountable / Interested Parties
Content package	Refers to any artefact in transit or at play time, including creatives, the assembled package that results after wrapping stages, or any associated schedule or playlist instructions that determine when and where the original creative will run. Ownership resolves by pathway: static or HTML = advertiser/agency; programmatic wrapper or assembled package = DSP/SSP, verifier or media owner according to custody; MO-dynamic = media owner; verification wrapper = verifier.	By pathway - see definition
Data feed	Runtime data source called by dynamic creatives at render time.	Third-party data provider (origin); DCO provider (binding); media owner (approves)
Inter-party transfer channel	The path moving packages or instructions between any two parties in the chain.	Both parties: sender originates; receiver authenticates

6.4 Vendor-shipped Code

Vendor-shipped code is software released by a third-party vendor that enters the delivery chain as a component of an approved package or is served at runtime. Vendor-shipped code is authored and maintained by a supplier whose update cycle is outside the media owner's direct control. Because the same vendor code may be deployed across many players or many participants simultaneously, a compromise or malicious update in vendor-shipped code can have an industry-wide blast radius. Controls referencing this asset apply primarily to the shipping vendor and to the media owners and other participants that receive and deploy it.

Asset	Definition	Accountable / Interested Parties
Vendor-shipped code	Versioned code shipped into the chain by any vendor, including CMS player software, verifier code, and DCO engine. May be bundled in an approved package or served at runtime.	The shipping vendor, e.g. CMS vendor, verification provider, DCO agency

7 Appendix — Third Party Attestation Questionnaire

This appendix provides the master question bank from which the OMA DOOH supply chain attestation questionnaires are drawn. It is intended to replace the existing third-party vendor questionnaires currently in use by OMA with questions that are aligned to the control requirements set out in this Guideline.

The question bank covers the following vendor categories: Dynamic content providers, Data platform providers, Verification providers, Programmatic providers (DSPs and SSPs), and CMS vendors. Each question is tagged to the vendor categories it applies to, so media owners can extract a tailored questionnaire for each supplier type. The 'Type' column indicates whether a question is new to this version of the questionnaire or has been updated from the previous version, to support media owners in prioritising re-attestation for existing suppliers.

Q#	Type	Response type	Question	Applies to
Administrative				
1	**Existing**	Free text	Company Name	Verification, DSP, SSP, Dynamic, Data, CMS vendor
2	**Existing**	Free text	Address	Verification, DSP, SSP, Dynamic, Data, CMS vendor
3	**Existing**	Free text	ABN	Verification, DSP, SSP, Dynamic, Data, CMS vendor
4	**Existing**	Free text	Description of the services you'll be providing to our mutual clients that involve your interaction with our systems	Verification, DSP, SSP, Dynamic, Data, CMS vendor
5	**Existing**	Free text	Your Name	Verification, DSP, SSP, Dynamic, Data, CMS vendor
6	**Existing**	Free text	Your Role	Verification, DSP, SSP, Dynamic, Data, CMS vendor
7	**Existing**	Free text	Your Email Address	Verification, DSP, SSP, Dynamic, Data, CMS vendor
8	**Existing**	Free text	Date you completed this questionnaire	Verification, DSP, SSP, Dynamic, Data, CMS vendor
Company assurance and governance				
9	**Existing**	Yes/No + evidence	Have you been operating for more than 5 years in Australia?	Verification, DSP, SSP, Dynamic, Data, CMS vendor

10	**Modified**	Yes/No + evidence	Do you hold a current SOC 2 report, ASAE 3150 report or equivalent independent assurance report?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
11	**Existing**	Yes/No + evidence	Can you share your certification documentation?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
12	**Existing**	Yes/No + evidence	Do you review your SOC 2 (ASAE 3150) or other data security policy at least annually?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
13	**Modified**	Yes/No + evidence	Have you experienced any cyber security incidents over the past 24 months?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
14	**Modified**	Yes/No + evidence	Have any other parties involved in your environment or service delivery experienced any cyber security incidents over the past 24 months?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
15	**Existing**	Yes/No + evidence	Do personnel who interact with media owner systems complete role-appropriate security awareness training?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
16	**Modified**	Yes/No + evidence	Do you hold Public Liability or Cyber Insurance that would respond to a cyber security breach arising from your platform or package?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
Content Integrity				
17	**Modified**	Multi-select + details	Which of the following do you use to ensure the HTML or JavaScript you provide does not contain malicious or unexpected code and cannot be tampered with while in your custody? Tick all that apply and add detail. Options: - Secure development lifecycle and peer code review - Unobfuscated, inspectable code provided for media owner review - Automated static analysis / malware scanning of packages - Integrity protection of packages at rest (access control, integrity monitoring) - Cryptographic signing or published hash so a recipient can verify the package is unchanged - Pre-publication review of content for unapproved or inappropriate material - Independent security testing of packages - Other (please describe)	Verification, DSP, SSP, Dynamic

18	**Modified**	Multi-select + details	For the content or code you provide, which of the following capabilities do you have? Tick all that apply and add detail. (Assessed at onboarding and at periodic reassessment, not per package.) Options: - You can declare, at onboarding, the origins and data endpoints your packages call at runtime - You can restrict packages to the declared endpoints (for example via an embedded Content-Security-Policy that denies all other loads) - Your packages run within the player rendering environment without requiring device filesystem, storage or operating-system access - You can constrain the browser or player capabilities your packages use to an agreed profile - You notify the media owner before adding or changing an endpoint or capability - You re-declare on material change and at periodic reassessment - You constrain non-visible image tags (for example tracking pixels) to a fixed small size or off-screen so they cannot affect layout or load unexpected content - Other (please describe)	Verification, DSP, SSP, Dynamic
19	**New**	Yes/No + evidence	For any data served to your creatives at render time, can each response be verified as originating from an approved source and unmodified, with a defined fallback where a response is anomalous?	Dynamic, Data
20	**New**	Yes/No + evidence	Does the player rendering environment you provide allow media owners to restrict creative capabilities (e.g. enforcing a Content-Security-Policy or sandbox) so creative behaviour can be bounded?	CMS vendor
21	**New**	List / free text	Beyond the items above, list any other controls (not already covered) you operate to ensure your content and code behaves only as approved and cannot introduce malicious or unapproved behaviour when it executes. Refer to the OMA guidelines for examples.	Verification, DSP, SSP, Dynamic, Data, CMS vendor
Delivery chain integrity				
22	**New**	Yes/No + evidence	Are updates to the CMS player software signed and verified by the player before installation?	CMS vendor

23	**New**	Yes/No + evidence	Do you operate a defined release-management process for the products and code you supply, including how security updates are versioned, released and communicated?	Verification, DSP, SSP, Dynamic, CMS vendor
24	**Modified**	Yes/No + evidence	Are the packages and code you ship signed at origin and built from approved source by a trusted process, and do you provide a means for the recipient to verify authenticity and detect modification before deployment (for example, independently published hashes, or a difference report against the last approved version)?	Verification, DSP, SSP, Dynamic, CMS vendor
25	**New**	Yes/No + evidence	At the point you deliver content or play instructions to the media owner, do you attest to, or provide a means to verify, the integrity of what is delivered, so the media owner can confirm it has not been altered within your platform before hand-off?	DSP, SSP
26	**New**	Multi-select + details	How do you manage the private keys used to sign your packages, code or data responses? Tick all that apply and add detail. Options: - Keys stored in an HSM or managed key vault - Documented key rotation schedule - Signing access restricted and logged - Defined key revocation and compromise-response process - Separate signing keys per environment or customer - Other (please describe)	Verification, DSP, SSP, Dynamic, CMS vendor
27	**New**	Yes/No + evidence	At each point where you wrap, modify or hand over a package, is the action attributable and recorded so the media owner can trace who changed what?	Verification, DSP, SSP, Dynamic
28	**New**	Yes/No + evidence	Does the player software protect cached content storage against unauthorised modification, for example, write-access restriction and integrity monitoring?	CMS vendor
29	**New**	List / free text	Beyond the items above, list any other controls (not already covered) you operate to assure the authenticity and integrity of content and code from origin to delivery. Refer to the OMA guidelines for examples.	Verification, DSP, SSP, Dynamic, Data, CMS vendor
Access and accountability				
30	**Existing**	Yes/No + evidence	Is multi-factor authentication enforced for all access to platforms involved in the solution, including front-end user access?	Verification, DSP, SSP, Dynamic, Data, CMS vendor

31	Modified	Yes/No + evidence	Are user access reviews performed every 90 days? (For example, ensuring that if a user of the platform has left their place of employment, they are removed)	Verification, DSP, SSP, Dynamic, Data, CMS vendor
32	Existing	Yes/No + evidence	Are accounts which have been idle for 90 days disabled for authentication for all parts of your platform?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
33	New	Yes/No + evidence	Do you collect, retain and monitor security logs across the systems used to deliver your services so that anomalous or malicious activity is detected and can be investigated?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
34	Modified	Yes/No + evidence	Do all platform and data-feed API endpoints enforce strong, modern authentication, and are data-feed responses served only over authenticated, transport-protected channels?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
35	New	List / free text	Beyond the items above, list any other controls (not already covered) governing access to the systems used to deliver your services and the accountability of actions. Refer to the OMA guidelines for examples.	Verification, DSP, SSP, Dynamic, Data, CMS vendor
Supply-chain assurance and resilience				
36	Modified	Yes/No + evidence	Will you notify the media owner of any suspected compromise affecting your package or platform within a defined window, and participate in an agreed incident-response and content-takedown process?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
37	New	Multi-select + details	Where your platform is shared across multiple customers, how do you enforce tenant isolation? Tick all that apply and add detail. Options: - Logical separation of each customer's data, content and configuration - Per-tenant access controls preventing cross-tenant access - Separate encryption keys per tenant - Isolation validated by testing (for example penetration testing) - Other (please describe)	Verification, DSP, SSP, Data, CMS vendor
38	New	Yes/No + evidence	Do you permit a right of audit and independent security testing of the services and code you provide, under non-disclosure?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
39	New	Describe / provide evidence	Do you perform penetration testing that covers the penetration-testing scope defined in the OMA guidelines? If yes, provide the most recent report or attestation. If your testing does not cover	Verification, DSP, SSP, Dynamic, Data, CMS vendor

			the recommended scope, specify what your scope does cover and why you chose not to test against the recommended scope.	
40	**New**	Yes/No + evidence	Are player software updates released in stages so that a faulty or malicious update can be detected and halted before it reaches your entire customer base?	CMS vendor
41	**New**	Yes/No + evidence	Do your contracts with sub-processors, downstream partners, and other parties involved in the delivery of your services include defined security obligations, a requirement for annual security attestation, and a right to terminate for material non-compliance?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
42	**New**	Yes/No + evidence	Do you perform security risk assessments of the sub-vendors and platforms involved in your creative preparation, delivery, or data services before onboarding them, and do you reassess on material change?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
43	**New**	List / free text	Beyond the items above, list any other measures (not already covered) by which you assure your security to partners, notify of compromise or change, and limit incident impact on shared infrastructure. Refer to the OMA guidelines for examples.	Verification, DSP, SSP, Dynamic, Data, CMS vendor
General security practices				
44	**New**	Yes/No + evidence	Do you maintain an accurate, current inventory of the assets and software used to deliver your services, including a process to detect and address unauthorised assets? (CIS Controls 1 and 2)	Verification, DSP, SSP, Dynamic, Data, CMS vendor
45	**New**	Yes/No + evidence	Do you operate a documented vulnerability management process and apply operating-system and application patches through automated patch management on a monthly or more frequent basis? (CIS Control 7)	Verification, DSP, SSP, Dynamic, Data, CMS vendor
46	**New**	Yes/No + evidence	Do you operate a documented change-management process so that changes to configuration, code and release pipelines are reviewed, approved and recorded? (CIS Control 4)	Verification, DSP, SSP, Dynamic, Data, CMS vendor
47	**New**	Multi-select + details	Which of the following do you apply to harden and protect the assets and software used to deliver your services? Tick all that apply and add detail. Options: - Secure configuration / hardening baseline (for example CIS Benchmarks)	Verification, DSP, SSP, Dynamic, Data, CMS vendor

			- Removal of unnecessary services, accounts and default credentials - Anti-malware / endpoint protection deployed and maintained - Application control or allow-listing - Automatic device lockout on user endpoints after failed authentication attempts (where supported) - Other (please describe)	
48	**Modified**	Yes/No + evidence	Do you encrypt customer data and DOOH delivery-chain data in transit across all relevant channels, including platform access, API access, data-feed responses, content package transfers, telemetry, and inter-party integrations?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
49	**Modified**	Yes/No + evidence	Do you encrypt customer data and DOOH delivery-chain data at rest wherever it is stored or cached, including platform databases, object storage, content and package repositories, logs, backups, and player or edge caches where applicable?	Verification, DSP, SSP, Dynamic, Data, CMS vendor
50	**New**	Yes/No + evidence	Do you follow a secure software development process, applying secure design principles to the code you ship into the delivery chain? (CIS Control 16)	Verification, DSP, SSP, Dynamic, CMS vendor
51	**New**	Yes/No + evidence	Do you maintain and periodically test backups and a documented data recovery process for the systems used to deliver your services? (CIS Control 11)	Verification, DSP, SSP, Dynamic, Data, CMS vendor
52	**New**	Multi-select + details	Which of the following general security practices do you maintain? Tick all that apply, add any others, and provide detail or evidence. Options: - Documented information security policy framework - Security governance with executive or board oversight - Risk management framework and risk register - Security awareness and training programme - Incident response plan (tested) - Business continuity and disaster recovery plan - Third-party / supply-chain risk management - Assessment or compliance to other good practices (e.g. NIST CSF, Essential 8) - Other (please describe)	Verification, DSP, SSP, Dynamic, Data, CMS vendor

8 Appendix — Technical Security Assessment Scopes

8.1 How to Use the Technical Security Assessment Scopes

The testing scopes in this Appendix are indicative starting points for each participant role. They are not prescriptive and each participant should adapt them, in consultation with OMA Cyber Committee, to reflect their specific environment, architecture, and deployment model. The intent is to provide a structured basis for briefing penetration testers and for responding to the relevant annual penetration testing question in the third-party attestation questionnaire (for relevant supply chain vendors). Where a participant's scope differs materially from the indicative scope provided here, they should document the rationale.

Each scope is aligned around the Framework's six threat scenarios: single screen compromise (S1), multi-screen compromise via operator infrastructure (S2), mass compromise via supplier platform (S3), data feed and CDN poisoning (S4), runtime behaviour divergence (S5), and supply chain package tampering (S6). The test objectives for each scope are mapped to the scenarios relevant to that participant's role, so testing is directed at the specific compromise patterns the Framework is designed to address rather than generic IT security coverage.

The scopes are written as test objectives rather than prescribed penetration testing approaches. Each objective in these scopes may require a different combination of approaches to address properly, and the right combination depends on the participant's specific environment and architecture. The intent is that participants and their testers determine the best approach to meet each objective, rather than fitting the engagement into a predefined category that may leave gaps.

Additionally, each participant should select the testing method that best suits their environment, risk profile, and the objectives being tested. Testing methods include:

- Black-box testing (no prior knowledge) is appropriate where exposure to an external or opportunistic attacker is the primary concern.
- Grey-box testing (limited credentials and architecture context) is typically the most efficient approach and a sensible default for most objectives.
- White-box testing (full access to credentials, configuration, and architecture) is appropriate where thorough validation of specific controls or maximum coverage within a fixed window is the priority.

Participants may use a combination of approaches across different test objectives. The scopes below do not prescribe a method; the right choice depends on what the participant is trying to validate and the maturity of their existing controls.

8.2 Penetration Testing Criteria

These criteria apply consistently across all vendor types (media owner, CMS platform vendor, DCO, SSP/DSP, verifier), regardless of engagement cost or vendor revenue. Test scope is calibrated to each party's risk profile as set out in the relevant assessment scope below.

8.2.1 Tester experience and competency

It is expected that each organisation determines that the testers engaged to perform the penetration tests are adequately competent. This includes checking for:

- Demonstrable experience in the specific test type required by the engagement (e.g. infrastructure, web application, mobile, source code review, cloud configuration) — if there is a single tester designated to the engagement whose background is exclusively web application testing is not an appropriate substitute for an infrastructure or source code review engagement, and vice versa.
- A track record of engagements of comparable scope and complexity, with references or prior work samples available on request.
- Where source code review is in scope, specific experience in secure code review (as distinct from dynamic/black-box testing) for the relevant language or stack.
- Recognised industry certifications relevant to the test type, used as supporting evidence rather than a standalone qualifier, for example CREST CRT/CCT, Offensive Security (OSCP/OSCE), GIAC (GPEN/GXPN/GWAPT) or equivalent. Vendors may propose testers without these where equivalent demonstrable experience is provided.

8.2.2 Entry criteria — before testing begins

The following must be agreed and confirmed before a penetration test commences:

- Scope is documented and agreed against the relevant assessment scope below e.g. systems, applications, IP ranges, source code repositories in scope, and explicit exclusions.
- Rules of engagement are agreed, including testing windows, permitted and excluded techniques (e.g. denial-of-service, social engineering), and escalation contacts for high-severity findings discovered mid-test.
- Written authorisation to test is in place from the asset owner.
- The test environment is confirmed representative of production, or any material differences are documented and their impact on result validity assessed.
- The proposed testing supplier and named tester(s), together with their relevant experience against the criteria above, are submitted to and confirmed by OMA Cyber Committee.
- The proposed format of the results documentation/attestation is submitted to and confirmed by OMA before testing starts.

8.2.3 Testing Methodology

The following should be made clear to the third party engaged to perform testing:

- Where a test objective depends on an attacker already holding some foothold e.g. a compromised player, a stolen or shared credential, a legitimate low-privilege tenant account, valid build/deployment access, the test must include an assumed-compromise phase starting from that position, in addition to any external testing. A report demonstrating only that an external, zero-access attacker could not gain initial entry does not constitute evidence against these objectives.
- Where scope includes a "representative sample", the sample size and selection criteria (e.g. spread of hardware/firmware versions, network configurations, site types, tenant environment etc.) must be specified and approved before testing, to prevent the sample being drawn only from the newest or best-configured sites/environments.

8.2.4 Exit criteria — on receipt of the report

A report should be treated as acceptable only where it demonstrates:

- Named tester(s) and testing firm, with methodology and dates stated.
- Scope tested matches the scope agreed at entry, with any deviations explained.
- Evidence of manual testing and exploitation — not solely automated scan output. This should include specifics such as request/response pairs, exploitation narratives, or business-logic findings that automated tooling cannot identify.
- Findings risk-rated against a stated methodology with business context, not a bare severity label.
- Specific, actionable remediation guidance rather than generic boilerplate.
- An executive summary pitched appropriately for a non-technical audience, separate from the technical detail.
- The final report must include an objective-by-objective coverage statement: for each test objective listed in the below scopes and state whether it was tested, the method used, and the outcome e.g. achieved, not achieved, or inconclusive. Objectives not tested must be stated as such with explanation.
- A stated approach and timeline for remediation verification/retest.

A report that fails several of these should be sent back rather than accepted as evidence of testing.

8.2.5 Acceptable attestation evidence

Where a full report cannot be shared for confidentiality reasons, a formal attestation letter from the testing firm is acceptable, provided it independently confirms: scope tested, dates, methodology, tester competency against the criteria above and a summary of the outcome of tested objectives, findings and risk ratings. The attestation must be signed by an authorised representative of the testing firm and not self-issued by the vendor being tested.

8.2.6 Unacceptable evidence

- Raw vulnerability scanner output (e.g. an unedited Nessus, Qualys, OpenVAS or Burp scan export) presented as a penetration test report. Automated scanning is a legitimate part of a proper test, but is not a substitute for one. It's unacceptable only where it is the entirety of the deliverable, with no manual verification or exploitation behind it.
- Reports with no named tester or firm, or where it's unclear whether testing involved a human analyst versus automated tooling alone.
- Self-attestation issued by the vendor being tested, with no independent third party involved.
- Undated reports, or reports where the tested scope cannot be matched to the assets in question.

8.3 Assessment Scope for Media Owners

8.3.1 Scope

The party defines the final in-scope and out-of-scope items for its engagement. The following are provided as examples to guide that definition.

8.3.1.1 Examples of in-scope items

- Player devices - hardware, enclosure, cabling and ports - at a representative sample of sites.
- Player operating system and standard image, and the local storage and content cache.
- Player rendering environment (the browser or engine that executes creative).
- The player network segment and its outbound (egress) paths.
- Fleet remote-management tooling used to administer the players.
- The content scheduling and CMS environment as accessed and configured by the media owner.

8.3.1.2 Examples of out-of-scope items

- Denial-of-service or load testing against live public screens, and display of test content on a live public screen.
- The effectiveness of the approval-time creative review process.
- Supply-chain custody between other parties, including wrap and assembly, inter-party transfer, and vendor build pipelines.
- The internal platform security of third-party-operated systems (CMS vendor platform, SSP, DSP, verifier servers, DCO platforms, third-party data providers), tested under those parties' own scopes.

8.3.2 Threat scenarios this party is seeking to prevent

- A single screen being tampered with - physically, or through local or network access - to display unauthorised content, expose credentials or keys, or serve as an entry point into the wider estate.
- A compromise that scales across the network, where stolen, shared or retained credentials, or the fleet management tooling, give an attacker control of content scheduling across many screens at once.
- Poisoning of a data feed or content source that a creative calls at display time, causing malicious or inappropriate content to render on screen.
- A creative that passed review behaving differently in production, with such divergence going undetected.

8.3.3 Test objectives

The objective of testing is to determine whether:

- A threat actor with physical or local access to a screen can place unauthorised content on it, recover credentials or keys from it, or use it as a foothold into the wider estate.
- A threat actor can obtain content-scheduling control across the network to schedule content.
- A threat actor can use the fleet remote-management tooling to reach or control players at scale.
- A threat actor can poison, intercept or substitute a data feed so that malicious or inappropriate content renders on screen.
- A threat actor can submit a creative that passes review but performs unapproved actions when it executes on a player.
- A threat actor can alter the content displayed on a screen without the divergence from the scheduled content being detected.
- A threat actor with a malicious package can break out of the player software to affect the underlying player device.

8.4 Assessment Scope for CMS Platform Vendor

8.4.1 Scope

The party defines the final in-scope and out-of-scope items for its engagement. The following are provided as examples to guide that definition.

8.4.1.1 Examples of in-scope items

- The hosted content management and scheduling platform - authentication, authorisation and session management.
- Multi-tenant isolation between media owner tenants on the platform.
- Platform audit logging and anomaly detection.
- The player software the vendor builds and ships, including the build and update distribution channel e.g. CI/CD pipeline.
- The player rendering environment the vendor ships - its runtime sandbox and bounding, and its feed-origin and transport enforcement.
- Package storage and ingestion within the platform.
- Supporting platform infrastructure, including administrative interfaces and APIs.

8.4.1.2 Examples of out-of-scope items

- Denial-of-service or load testing against production platforms, unless separately agreed.
- Content approval decisions made by media owners.
- Other parties' platforms (SSP, DSP, verifier servers, DCO platforms, third-party data providers).
- Media owners' own platform configuration and credential practices, and their physical player estates.

8.4.2 Threat scenarios this party is seeking to prevent

- A breach of the hosted platform reaching across media owner tenants.
- Compromise of the underlying infrastructure and platform allowing an attacker to gain scheduling control of an owner's or multiple owners' network.
- A forged or malicious player-software update being injected into the build or distribution channel and reaching the fleet.
- The player software failing to confine creative behaviour or feed calls at runtime, allowing malicious or poisoned content to render.
- A package being tampered with in platform storage or in transit and accepted without integrity checking.

8.4.3 Test objectives

The objective of testing is to determine whether:

- A threat actor can cross tenant boundaries on the hosted platform to read or affect another media owner's schedules, content or data.
- A threat actor can obtain scheduling control of a media owner's network due to weaknesses in the underlying infrastructure and platform.
- A threat actor can inject a forged or malicious update into the player-software build or distribution channel such that it reaches a media owner's player fleet.
- A threat actor can modify a package held in platform storage, or substitute one in transit, without it being detected at ingestion.

8.5 Assessment Scope for DCOs

8.5.1 Scope

The party defines the final in-scope and out-of-scope items for its engagement. The following are provided as examples to guide that definition.

Note: if a third party CDN is used to serve content to the media owner, refer to supplementary scope to include as part of the penetration testing scope.

8.5.1.1 Examples of in-scope items

- The platform code-update process for deployed creatives.
- The build pipeline producing creatives, and the origin signing of produced creatives.
- Feed-binding and deployment validation processes.
- The data feeds, CDN and any runtime HTML-writing engines the DCO operates.

8.5.1.2 Examples of out-of-scope items

- Media-owner-side runtime enforcement and content approval decisions.
- The player estate, the CMS, and other parties' platforms.

8.5.2 Threat scenarios this party is seeking to prevent

- Poisoning of a data feed, CDN or runtime-writing engine the DCO operates, causing malicious or inappropriate content to render.
- A misconfiguration binding a creative to the wrong or test feed.
- A silent post-approval code update changing deployed creative behaviour.
- The build pipeline being compromised, or a produced creative being tampered with before signing.

8.5.3 Test objectives

The objective of testing is to determine whether:

- A threat actor can poison or compromise a data feed, CDN, cloud environment or runtime-writing engine operated by the DCO so that malicious or inappropriate content renders on screen.
- A threat actor, or a misconfiguration, can bind a creative to the wrong or test data feed and have it deploy without validation.
- A threat actor can push a post-approval code update that changes deployed creative behaviour without it being declared or detected.
- A threat actor can compromise the build pipeline, or tamper with a produced creative before it is signed, without detection.

8.6 Assessment Scope for SSPs/DSPs

8.6.1 Scope

The party defines the final in-scope and out-of-scope items for its engagement. The following are provided as examples to guide that definition.

Note: if a third party CDN is used to serve content to the media owner, refer to supplementary scope to include as part of the penetration testing scope.

8.6.1.1 Examples of in-scope items

- The SSP or DSP platform used to receive, store, process, approve, bid on, select, wrap, transform or deliver creatives and play instructions.
- Creative ingestion, validation, storage, wrapping, transformation and delivery workflows, including any runtime-loaded code, tags, templates or package components.
- Bid request, bid response, campaign configuration and play-instruction workflows that can influence which content is displayed and where it is displayed.
- The platform code-update process, build pipeline, release process and deployment controls for code that can affect creative selection, wrapping, delivery or runtime behaviour.
- Content delivery paths, CDN configuration, storage buckets, APIs and transfer channels used to move creatives, wrappers, tags, metadata or play instructions between parties.

8.6.1.2 Examples of out-of-scope items

- Media-owner-side runtime enforcement, player configuration and content approval decisions.
- The player estate, CMS platform internals, verifier platforms, DCO platforms, third-party data platforms and other parties' systems, except where integration points are being tested.
- Denial-of-service or load testing against production bidding, delivery or reporting platforms, unless separately agreed.
- Advertiser, agency or media owner environments outside the SSP or DSP platform boundary.

8.6.2 Threat scenarios this party is seeking to prevent

- A compromise of the SSP or DSP platform allowing an attacker to submit, approve, select, wrap or deliver malicious or inappropriate content across one or more media owners.
- A breach of a shared platform crossing tenant boundaries and exposing or altering another customer's creatives, campaigns, schedules, reporting data, credentials or configuration.
- A malicious or unauthorised change to a creative, wrapper, bid response, campaign configuration or play instruction being accepted and delivered without detection.
- A runtime-loaded component, tag, CDN path or platform update changing creative behaviour after approval or causing content to execute differently in production.
- A compromised build, release or deployment pipeline causing malicious platform code, wrapper code or delivery logic to affect multiple campaigns or media owners.

8.6.3 Test objectives

The objective of testing is to determine whether:

- A threat actor can submit, modify, wrap, approve, select or deliver a creative or play instruction in a way that results in unauthorised content being displayed.
- A threat actor can alter a stored creative, wrapper, tag, metadata item, bid response or play instruction without the change being attributable or detected.
- A threat actor can cause runtime-loaded code, tags or CDN-hosted components controlled by the SSP or DSP to execute behaviour that differs from what was reviewed or declared.

A threat actor can compromise the platform build, release or deployment process so that malicious code or delivery logic is introduced into production without detection.

8.7 Assessment Scope for Verifiers

8.7.1 Scope

The party defines the final in-scope and out-of-scope items for its engagement. The following are provided as examples to guide that definition.

Note: if a third party CDN is used to serve content to the media owner, refer to supplementary scope to include as part of the penetration testing scope.

8.7.1.1 Examples of in-scope items

- The verification mechanism technology stack.
- Package storage and ingestion.
- Any wrapping or assembly the verifier performs on creatives.
- The build pipeline producing assembled packages.
- The verification delivery architecture - whether measurement code is packaged locally or fetched from a server at runtime, and that server.

8.7.1.2 Examples of out-of-scope items

- Media-owner-side runtime enforcement at the player.
- Content approval decisions.
- The player estate, the CMS, and other parties' platforms.

8.7.2 Threat scenarios this party is seeking to prevent

- A compromise of the server from which measurement code is fetched at runtime, pushing malicious code to every screen running the tag across owners.
- A package executing runtime code that differs from what was reviewed, or a silent post-approval update changing behaviour.
- The build pipeline being compromised so that every release carries a payload.
- A package being tampered with during wrapping, in storage, or in transit, without detection.

8.7.3 Test objectives

The objective of testing is to determine whether:

- A threat actor can compromise the server from which the package fetches code at runtime and push malicious code to every screen running the verifier package across owners.
- A threat actor can cause the wrapper to execute runtime code that differs from what was reviewed.
- A threat actor can push a post-approval code update that changes deployed wrapper behaviour without it being declared or detected.
- A threat actor can compromise the build pipeline so that released verification packages carry a malicious payload.
- A threat actor can inject an undetected change while a creative is wrapped or assembled.
- A threat actor can modify a stored or in-transit package without it being detected.

8.8 Supplementary Scope — Third Party CDN and Content-Delivery Endpoints

8.8.1 When this applies

This supplementary scope applies to any party that relies on a third-party content delivery network (CDN) or content-delivery endpoint to serve dynamic creative content, data feeds, or media assets, whether or not that party operates the CDN infrastructure itself. Where this applies, the relying party must include this supplementary scope as part of its own penetration testing engagement, in addition to (not instead of) its primary assessment scope above.

This scope tests the relying party's integration with and configuration of the CDN or delivery endpoint. It does not require, and cannot practically obtain, direct testing of the underlying CDN provider's core infrastructure.

8.8.2 Scope

The party defines the final in-scope and out-of-scope items for its engagement. The following are provided as examples to guide that definition.

8.8.2.1 Examples of in-scope items:

- The relying party's configuration of the CDN or delivery endpoint, including origin authentication, access controls, cache and purge controls, and storage/bucket permissions.
- The transport protection and response authenticity of the data feed or CDN endpoint served to creatives (e.g. TLS configuration, signed URLs, short-lived tokens).
- The relying party's runtime handling of that endpoint's responses, where not already covered under the party's primary scope.
- Supporting infrastructure the relying party controls for the endpoint (e.g. WAF rules, DNS configuration under the relying party's control).

8.8.2.2 Examples of out-of-scope items:

- The CDN provider's own core platform, infrastructure or multi-tenant architecture — addressed through the provider's own third-party assurance evidence, not direct testing.
- Player estates, the CMS, and other parties' platforms, except where integration points with the CDN are being tested.
- Denial-of-service or load testing against the CDN provider's production infrastructure.

8.8.3 Threat scenarios this party is seeking to prevent

- The served endpoint being compromised or substituted so that it returns attacker-controlled data or content to players.
- Responses being intercepted or tampered with in transit e.g. through CDN cache poisoning, DNS attack, or man-in-the-middle, so that poisoned data renders on screen.
- Misconfigured access controls or storage permissions on the relying party's side allowing unauthorised modification of content before it reaches the CDN or is served from it.

8.8.4 Test objectives

The objective of testing is to determine whether:

- A threat actor can compromise or substitute the served endpoint, within the boundary of the relying party's configuration, so that it returns attacker-controlled data to players.
- A threat actor can intercept or tamper with responses in transit so that poisoned data renders on screen.
- A threat actor can strip or bypass transport protection or response authenticity so that a forged response is accepted.

- A threat actor can exploit misconfigured storage, cache, or access controls under the relying party's control to modify or substitute content before it is served.

---
end