Network Working Group
Internet-Draft
Intended status: Informational
Expires: 17 September 2026

J. Chen
Tsinghua University

16 March 2026

Origin-Bound Validation for HTTP Server-Initiated Delivery
draft-chen-httpbis-server-delivery-origin-boundary-00

Abstract

This document describes origin-binding considerations for HTTP server-initiated delivery mechanisms that can cause a user agent to associate a delivered representation with an origin other than the origin that established the underlying transport connection. The motivation is a class of cross-origin attacks demonstrated against HTTP/2 server push and Signed HTTP Exchange (SXG), in which a server that is authorized by a shared TLS certificate for multiple Subject Alternative Name (SAN) entries can cause content to be accepted under the authority of a different origin.

This document provides security guidance for user agents, origin servers, intermediaries, and deployment operators. In particular, it recommends that user agents reject server-initiated deliveries whose asserted authority is not origin-consistent with the active request context, and that implementations avoid using multi-domain shared certificates as a basis for SXG attribution across unrelated origins. It also outlines operational considerations for certificate lifecycle management where shared certificates are unavoidable.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 17 September 2026.

Table of Contents

1. Introduction

The web security model relies on origin separation. In browsers, the Same-Origin Policy (SOP) is generally defined over the tuple of scheme, host, and port. HTTP, however, also contains the notion of authority, and HTTP/2 and HTTP/3 allow a single connection to be considered authoritative for multiple origins when the server certificate authenticates those origins. This difference is normally constrained by request routing rules, but server-initiated delivery features create cases in which the origin ultimately associated with a resource can diverge from the origin that established the connection.

Recent research demonstrated that this divergence is exploitable in practice when three conditions align: first, the server is authenticated by a certificate whose SAN list covers multiple domains; second, the delivery mechanism allows the sender to assert or imply an origin distinct from the current connection origin; and third, the user agent accepts the delivered object as if it were same-origin with the asserted origin. Under those conditions, an attacker who controls one domain in a shared certificate can target another domain covered by the same certificate. [NDSS25-CROSSPUSH] reports that the attack surface includes HTTP/2 server push and SXG, and that the resulting impact can include cross-site scripting, cookie manipulation, phishing, and malicious file delivery.

This document turns those findings into implementation and deployment guidance. Its design goal is conservative: server-initiated delivery ought not relax origin boundaries beyond those already accepted for ordinary same-origin fetches.

2. Requirements Language and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

For the purposes of this document, _connection origin_ means the origin that the user agent is actively communicating with on a given HTTP connection for the request context that triggered a server-initiated delivery. _asserted origin_ means the origin under which a delivery mechanism asks the user agent to store, attribute, execute, or otherwise process a delivered representation. _shared certificate_ means a TLS certificate whose SAN extension contains more than one host name spanning more than one registrable domain, or host names administered by distinct security principals.

3. Problem Statement

HTTP semantics define origins and authorities for different purposes. HTTP/2 and HTTP/3 permit connection reuse and server authority across names covered by the authenticated certificate. The cited research shows that this transport-layer flexibility becomes security-significant when a delivery feature lets the sender override, or appear to override, the origin under which content is later processed. In the attack model, the browser effectively accepts a resource that would be cross-origin under the traditional SOP as though it were same-origin because the transport authority check is satisfied by the SAN list; see [NDSS25-CROSSPUSH].

Two mechanisms are especially relevant:

* HTTP/2 and HTTP/3 server push can indicate the target authority of a pushed request using pseudo-header fields or equivalent request metadata, and some implementations have accepted such pushes when the asserted authority matches a host authenticated by the certificate for the connection; see [RFC9113], [RFC9114], and the discussion in [NDSS25-CROSSPUSH].

* SXG allows a user agent to attribute a delivered response to the request URL after signature validation, decoupling attribution from the immediate network origin. [NDSS25-CROSSPUSH] and [SXG-MAIN] describe how a shared certificate can cause that attribution boundary to expand to other names covered by the certificate.

The root causes are therefore: (a) a mismatch between strict URI-origin security boundaries and looser certificate-authority boundaries; and (b) weak alignment between certificate possession and current domain ownership in deployment practice, including resale, takeover, or delayed revocation cases; see [NDSS25-CROSSPUSH].

4. Threat Model

This document assumes an attacker who does not need to intercept the victim’s traffic and does not need a network-path position. The attacker controls at least one origin that can legitimately terminate TLS with a certificate whose SAN list also covers one or more victim origins. The research describes several practical paths to that condition, including domain resale, dangling-domain takeover, validation reuse, grace-period abuse, and failures of certificate revocation workflows; see [NDSS25-CROSSPUSH].

Once in possession of such a certificate, the attacker causes a user agent to visit an attacker-controlled origin and then supplies a server-initiated delivery that is attributed to a victim origin named in the same certificate. The user agent’s incorrect origin association is the primary failure. The attacker can then exploit either the response body or response headers, including script payloads, Set-Cookie, Strict-Transport-Security, or Content-Disposition, to affect the victim origin’s security context or user trust decisions, as discussed in [NDSS25-CROSSPUSH].

5. Security Design Principles

User agents and specifications implementing server-initiated delivery MUST preserve the invariant that transport-layer authorization is not by itself sufficient to establish web-origin equivalence. A certificate that authenticates multiple names proves only that the presenter can terminate TLS for those names at validation time; it does not prove that the presenter is entitled to inject same-origin active content for every name in the certificate.

Accordingly, any feature that permits a sender to nominate or imply an origin for later processing MUST apply an origin-consistency check that is at least as strict as the check applied for an ordinary same-origin fetch initiated by the protected origin itself.

6. User Agent Guidance

6.1. General Guidance

A user agent SHOULD NOT store, execute, render, or otherwise attribute a server-initiated delivery to an asserted origin unless that asserted origin is validated according to the mechanism-specific guidance in this document.

If mechanism-specific validation fails, the user agent SHOULD discard the delivered representation, SHOULD treat it as unusable for cache reuse, and SHOULD generate a diagnostic signal suitable for developer tools or telemetry.

6.2. Guidance for HTTP/2 and HTTP/3 Server Push

For HTTP/2 and HTTP/3 server push, the user agent SHOULD verify that the asserted authority of a pushed request exactly matches the origin context that authorized the push. At minimum, the scheme, host, and port of the pushed request ought to be same-origin with the request that created the push context.

A user agent SHOULD reject a pushed response when the pushed request names a host that differs from the host of the associated client-initiated request, even if the connection certificate is valid for both hosts.

A rejected push SHOULD NOT be inserted into any HTTP cache, preload cache, memory cache, or speculative fetch cache under either the asserted origin or the connection origin.

When connection coalescing or origin coalescing is used, an implementation MAY perform an additional confirmation step before accepting server push for a coalesced origin. For example, an implementation might confirm that the asserted origin resolves consistently with the established connection before accepting a push for that origin. This document does not require a specific confirmation algorithm.

6.3. Guidance for Signed HTTP Exchange

A user agent processing SXG SHOULD ensure that attribution of a signed exchange to the request URL is not based solely on the fact that the signing certificate is valid for multiple names.

For the purposes of this document, a certificate can be treated as origin-bound for attribution only when the implementation can establish that the certificate identifiers relevant to validation correspond to a single publisher context or to an explicitly configured equivalent-control policy. A general-purpose multi-domain shared certificate is not a safe default basis for attribution across unrelated origins.

A user agent SHOULD reject or otherwise decline publisher-origin attribution for SXG processing when the signing certificate authenticates multiple unrelated registrable domains and the implementation cannot establish explicit same-administrator equivalence among them.

Implementations MAY support local policy exceptions for tightly coupled deployments, but such exceptions SHOULD be disabled by default, SHOULD be auditable, and SHOULD NOT be inferred solely from SAN co-membership.

7. Origin Server, CDN, and Intermediary Guidance

An origin server, CDN, or intermediary that generates server push SHOULD NOT generate a pushed request whose asserted authority differs from the authority of the client-initiated request that created the push context.

An origin server, CDN, or SXG packaging service SHOULD NOT use a general-purpose shared certificate to sign exchanges intended for attribution as publisher-origin content across unrelated origins.

Where operationally feasible, services that deploy server push or SXG SHOULD use dedicated single-origin certificates for the affected origin, even when the underlying TLS termination infrastructure supports larger shared certificates for other purposes.

Implementations SHOULD log attempted cross-authority pushes and rejected shared-certificate SXG signing operations as security events.

8. Certificate Lifecycle and Registration Guidance

8.1. Registrars and Domain Transfer Operators

Domain transfer workflows should warn that existing certificates, including certificates visible in Certificate Transparency logs, can outlive a transfer and may still be usable in server-initiated delivery attacks if revocation is not completed. This recommendation follows the observation in [NDSS25-CROSSPUSH] that domain buyers should inspect Certificate Transparency logs when registering or acquiring domains that might previously have been covered by shared certificates.

Registrars should provide a simple mechanism for notifying the gaining registrant about recent certificates for the transferred domain and should provide links or guidance for revocation assistance.

8.2. Certificate Authorities

Certificate Authorities should offer an authenticated process by which a current domain controller named in a shared certificate can request removal of that domain from future reissuance and can request investigation of continued misuse.

When a CA receives credible evidence that one SAN entry in a shared certificate is no longer under the control of the entity presenting the certificate, the CA should prioritize revocation or replacement workflows that preserve service continuity for unaffected domains while eliminating the stale binding for the affected domain.

CA tooling should make it practical to replace a shared certificate with a narrower certificate that omits disputed SAN entries, and should notify all known certificate subscribers when such a narrowing action is initiated.

9. Deployment Considerations

Some deployments historically used HTTP/2 or HTTP/3 connection coalescing and server push to optimize performance across related hostnames. The guidance in this document intentionally trades away some of that flexibility to restore a crisp origin boundary. Deployments that rely on such optimizations should migrate to alternatives that do not depend on cross-origin attribution, such as

preconnect, preload, 103 Early Hints, or explicit same-origin fetches initiated by the protected origin.

Organizations that currently use shared certificates across independent business units, subsidiaries, tenants, or customer domains should treat that arrangement as a security-risk multiplier for any active-content delivery feature. In those environments, certificate partitioning is the preferred mitigation.

10. IANA Considerations

This document has no IANA actions.

11. Security Considerations

This entire document is about security. The main security objective is to prevent the transport authorization conferred by a TLS certificate from being incorrectly elevated into same-origin execution authority across unrelated origins. Failure to apply the checks and deployment precautions described here can permit active-content injection, phishing under a trusted origin indicator, manipulation of origin-scoped cookies and transport policies, and high-trust malicious download scenarios. [NDSS25-CROSSPUSH] reports concrete examples, including attacks against Microsoft-associated domains and broad exposure across browsers and domains covered by shared certificates.

The mitigations here do not solve all certificate misuse, nor do they eliminate risks from compromised same-origin servers. They are intended only to restore the principle that server-initiated delivery features ought not weaken origin isolation relative to ordinary fetch processing.

12. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC9113] Thomson, M. and C. Benfield, "HTTP/2", RFC 9113, June 2022, <https://www.rfc-editor.org/info/rfc9113>.

[RFC9114] Bishop, M., "HTTP/3", RFC 9114, June 2022, <https://www.rfc-editor.org/info/rfc9114>.

[SXG-MAIN] Yasskin, J., "Signed HTTP Exchanges", 2019, <https://wicg.github.io/webpackage/draft-yasskin-http-origin-signed-responses.html>.

13. Informative References

[NDSS25-CROSSPUSH] Chen, P., Chen, J., Zhang, M., Wang, Q., Zhang, Y., Xu, M., and H. Duan, "Cross-Origin Web Attacks via HTTP/2 Server Push and Signed HTTP Exchange", NDSS Symposium 2025, February 2025, <https://www.ndss-symposium.org/ndss-paper/cross-origin-web-attacks-via-http-2-server-push-and-signed-http-exchange/>.

Author’s Address

Jianjun Chen
Tsinghua University
China
Email: jianjun@tsinghua.edu.cn