**Computer Fraud and Security**
**ISSN (online): 1873-7056**

# A Layered Revocation Resilience Model: CRL, OCSP, and Bounded Fail-Open Policy in Internet-Scale PKI

**Naresh Charugundla**
Independent Researcher, USA

**Abstract**

Certificate revocation is a critical yet structurally fragile component of Public Key Infrastructure. While cryptographic mechanisms for revoking trust are well-specified, their real-world effectiveness depends on factors that extend well beyond protocol design, including network availability, infrastructure reachability, client-side implementation behavior, and the operational realities of distributed systems at Internet scale. This article examines certificate revocation through a resilience-oriented lens, proposing a layered model that integrates Certificate Revocation Lists, the Online Certificate Status Protocol, and bounded fail-open behavior as complementary trust signals rather than competing alternatives. In the revocation architecture, every mechanism plays an important role and contributes a distinct and non-substitutable property. For instance, CRL provides durable and cacheable baseline coverage that is consistent across infrastructure interruptions. OCSP delivers high temporal fidelity when connectivity permits, and with stapling, latency and responder dependency can further be reduced. Bounded fail-open behavior manages remaining uncertainties with the help of policy-driven eligibility conditions instead of silent defaults. Together, these layers enable trust evaluation to degrade gracefully under partial failure conditions. The model reframes revocation from a binary enforcement problem into a resilience challenge, one in which the objective is not to eliminate uncertainty but to make it explicit, bounded, and manageable. The architectural trade-offs among security enforcement, availability continuity, and trust predictability are examined as first-class design considerations relevant to modern PKI deployments.

**Keywords:** Certificate Revocation Infrastructure, Public Key Infrastructure Resilience, Online Certificate Status Protocol, Trust Signal Degradation, Fail-Open Validation Policy

## 1. INTRODUCTION

Public Key Infrastructure (PKI) is a key enabling technology for secure digital communications, providing authentication, confidentiality, and integrity via certificates [1, 2]. The CAs bind cryptographic key pairs with verified identities and then issue certificates. Key management and revocation procedures keep PKI systems trustworthy over the entire life of a certificate [1]. Web security, along with the healthcare, finance, IoT and government service sectors, relies heavily on these digital ecosystems. Many operations are more demanding than ever, like certificate revocation, which has historically been one of the most failure-prone operations in PKI [3]. A CA may technically revoke a credential, but that revocation message may never be observed reliably by a relying party that is verifying that credential. The problem is not fundamentally cryptographic but architectural and systemic. Out-of-band revocation, such as Certificate Revocation Lists or the Online Certificate Status Protocol, may also be available, but it depends on network conditions, whether any supporting infrastructure is reachable, and how the client software is configured and implemented [17].

Nevertheless, even where the aforementioned mechanisms are implemented, there is often an important gap between the theoretical promise of revocation and its practical implementation, as most work on revocation addresses CRLs, OCSP, and fail-open as alternate policies, rather than as complementary mechanisms that provide non-interchangeable properties for revocation. This view hides the architectural relationship between these mechanisms, and the extent to which revocation failures can be diagnosed, predicted, and avoided. There is currently no general architectural design for how to compose, order, and gracefully degrade these mechanisms to trade off between security and availability given the reality of partial failures.

This article contributes to this research gap by proposing the layered trust revocation resilience model in form of a hierarchy of confidence-ordered trust. Thus, CRLs, OCSP and bounded fail-open behavior are put into the context of their contribution to the evaluation of trustworthiness in case of failures. This paper is guided by three research questions: What unique and non-substitutable properties do different revocation mechanisms contribute

to trust evaluation in a failure scenario? In what cases and in what order should multiple revocation signals be issued to realize graceful degradation instead of an abrupt loss of trust? (3) What makes bounded fail-open a reasonable architectural resilience state, and who decides? The article proposes a layered revocation resilience model, where CRLs, OCSP, and bounded fail-open behavior are seen as complementary forms of trust signals. This layered model is designed to implement a structured degradation path and the continuity of trust evaluation, especially when individual mechanisms would otherwise not be acceptable.

## 2. REVOCATION AS A RELIABILITY PROBLEM IN INTERNET-SCALE PKI

### 2.1 The Gap Between Revocation State and Revocation Visibility

The conceptual simplicity of certificate revocation belies its operational complexity. When a CA marks a certificate as revoked, that state exists authoritatively in one place: the issuing system. Every subsequent step in the process, including CRL generation, distribution point serving, OCSP responder synchronization, and client-side caching, introduces opportunities for divergence between that authoritative state and what relying parties observe at validation time [17]. CRL update intervals are measured in hours or days. OCSP responses carry validity windows that persist well beyond the moment of query. Cached revocation data may remain in use across multiple validation events without refresh. In each case, a relying party operates on a version of revocation truth that is temporally displaced from the actual state maintained by the CA [14].

Coordinating issuance, renewal, and revocation procedures across distributed systems is necessary for PKI certificate management. As defined in RFC 5280, CRLs provide a periodically published and signed list of revoked certificates [17]. This enables offline verification without real-time connectivity. However, CRLs are concerned with the challenges associated with distribution overhead and staleness, especially in large-scale environments where revocation events must occur quickly [14]. Internet-wide active TLS scanning has revealed that CRLs are experiencing a measurable revival in deployment, with a growing proportion of certificates referencing CRL distribution points alongside or in place of OCSP endpoints, suggesting that operators are reassessing the relative reliability of the two mechanisms under real-world network conditions [25]. By recognizing the gap of structural property in distributed trust systems, design problems can be avoided. Furthermore, architects must focus on improving delivery reliability, cache coherence, and failure propagation instead of simply optimizing theoretical freshness. To address these distributed system concerns, distributed systems thinking is necessary [5, 7].

### 2.2 Revocation Under Partial Connectivity and Heterogeneous Clients

In Internet-scale systems, revocation mechanisms must operate under conditions of partial connectivity, infrastructure outages, and heterogeneous client behavior [6]. A relying party's ability to determine revocation status may be affected by transient network failures, delayed propagation of revocation data, or access constraints that prevent direct communication with revocation endpoints. These conditions introduce uncertainty even when revocation information exists and is technically correct [17]. The deployment of cryptographic mechanisms in Internet-scale systems has repeatedly demonstrated that even well-designed protocols face operational friction when the infrastructure supporting them is unavailable or inconsistent. DNSSEC, for instance, encountered significant challenges arising from the large scale and distributed nature of DNS, data caching behavior, and heterogeneous operations across autonomous administrations. These lessons apply equally to PKI revocation infrastructure [7].

Client-side behavior introduces additional variability. Some TLS implementations treat OCSP errors as severe failures, while others silently continue. Some cache revocation data aggressively; others do not cache it at all. This inconsistency means that revocation effectiveness is not a uniform property across the PKI ecosystem. It is a distribution of behaviors across heterogeneous implementations, each responding differently to the same infrastructure conditions [9]. A comprehensive survey of certificate revocation behavior across the TLS ecosystem found that this variability is not incidental but structurally embedded: client implementations differ substantially in how they handle OCSP unavailability, with some silently accepting certificates when the responder is unreachable and others enforcing rejection, producing divergent trust outcomes for identical failure conditions across the same client population [28]. It is reflected in the browser-specific policies, like the ones published by Chromium and Mozilla, which enforce revocation checking while permitting rational exceptions to ensure availability and better user experience [23, 24].

### 2.3 Framing Revocation as a Resilience Challenge

Traditional revocation approaches often assume that revocation information will be available when needed and that failure to retrieve it should result in immediate rejection. This assumption simplifies the validation logic; however, large-scale deployments cannot avoid availability constraints [15, 17]. The security control function of revocation should not be treated as the only security fail-closed mechanism because this approach will create fundamental system weaknesses that turn temporary infrastructure failures into major trust crises. The issue has received recognition in existing research about API gateways and security systems because their default fail-open and fail-closed settings depend on the importance of their endpoints instead of applying standardized rules [4].

Recognizing revocation as a reliability problem motivates a shift in how revocation mechanisms are structured. Instead of relying on a single source of revocation truth, resilience-oriented approaches combine multiple signals with differing availability and freshness characteristics [5, 13]. This shift does not weaken cryptographic trust. Rather, it acknowledges that trust continuity depends on the ability to make informed decisions even when ideal conditions are not met [14]. This reliability perspective serves as the foundation for the proposed layered revocation resilience model in this article. By integrating CRLs, OCSP, and bounded fail-open behavior as key layers, the model ensures trust evaluation under partial failure conditions while making revocation uncertainty manageable [30,31].

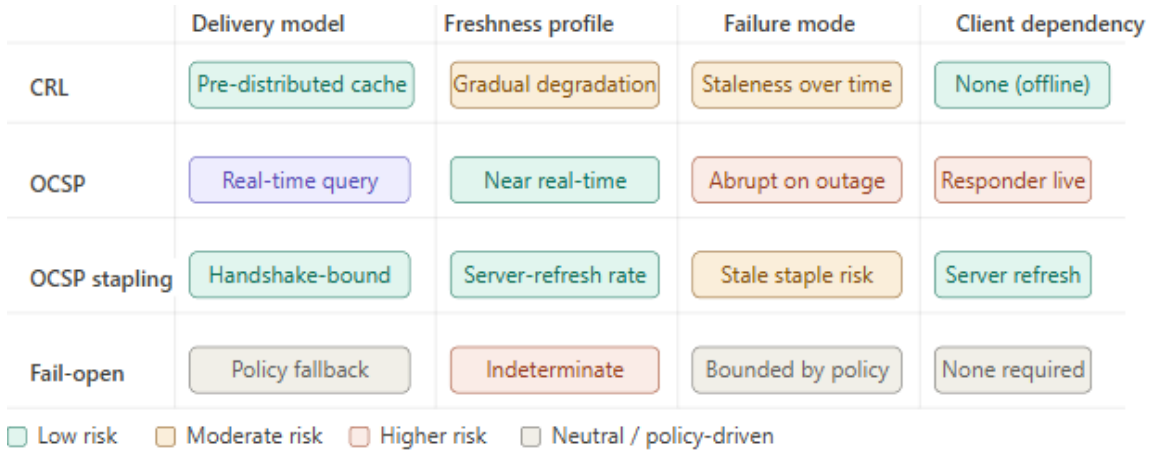


Figure 1: Revocation Mechanism Reliability Profile [17, 18, 22]

## 3. ARCHITECTURAL OVERVIEW OF REVOCATION MECHANISMS

### 3.1 CRLs: Distribution-Oriented Architecture

In the distribution model, revocation state information is published by a CA at regular intervals and made available at the distribution points listed in the certificate's CRL Distribution Points extension (RFC 5280) [17]. This model relies on distribution and caching of revocation information, providing the relying party with revocation state information before the validation process begins. CRLs strike a balance between freshness and availability by storing information even in the event of a temporary link failure. However, they still incur a delay between when a revocation event takes place and when it is known globally [14].

Practical scalability refers to the limitations of CRL sizes. Full CRLs are linear in size with the number of revoked but unexpired certificates, which forces verifying parties to incur downloading and parsing costs. Delta CRLs, as defined in RFC 5280, partially solve this problem, because they only distribute changes from the most recently published base CRL [17]. Delta CRLs introduce new dependencies, however, on coordinating delta issuance schedules and on CRL clients maintaining a base CRL state between uses. In high-throughput environments, such as smart grid networks or IoT-connected devices, CRL distribution cost is a substantial constraint on operation [13, 16].

A further benefit of CRLs is their well-defined degradation behavior (absent a refresh, confidence in revocation information diminishes slowly over time), allowing the choice to rely imperfectly. This is seen as a plus in the layered model [14], but CRLs alone lack the properties required for a modern Internet-scale PKI. A disadvantage of any periodic update model is latency, and the growing number of PKI deployments in different domains such

as web security, government services, and IoT has put increasing load on revocation infrastructures that distribution-only methods cannot relieve [13].

### 3.2 OCSP: Query-Oriented Architecture and Stapling

To support real-time revocation checking, OCSP uses a query-based approach, wherein certificate validation is achieved by generating on-demand queries to stateless OCSP responders [18]. OCSP reduces transfer size usage when compared to CRLs, while supporting near real-time revocation checking. OCSP is subject to network latency and availability, as for each OCSP validation request there is the need to talk to a networked server [18, 21]. The Lightweight OCSP profile addresses this issue for high-volume scenarios by allowing caching and reducing the load on the OCSP responders [19].

OCSP stapling, as specified in RFC 6066, is used in modern handshake protocols such as TLS 1.3. In order to overcome performance and privacy weaknesses of having the client drive OCSP requests, the TLS server pre-fetches the OCSP response and provides it as part of the TLS handshake [20]. This prevents latency, and endpoint websites do not learn the browsing behavior of the users. It can also be more reliable when access to the internet from a local environment is restricted [20, 21]. Other research solutions that propose alternative OCSP architectures in PKI systems for smart grids and energy storage have proved that this can be accomplished to meet freshness and availability requirements in resource-constrained environments [13].

OCSP failure modes are orthogonal to the CRL failure modes, including the capacity of the OCSP responder, geographic routing failures, and failures to validate a certificate chain. The most meaningful disparity is that while a stale CRL is increasingly less trustworthy the longer it has been cached, an unreachable OCSP responder immediately and unconditionally terminates the availability of information for relying parties. This condition, like certificate validation, may cause relying parties to either terminate validation or continue without confirmed revocation status to avoid reliance on third parties [9, 15]. The CA/Browser Forum Baseline Requirements and browser policy codify this tension by requiring revocation checking, with some exceptions, to ensure availability [22].

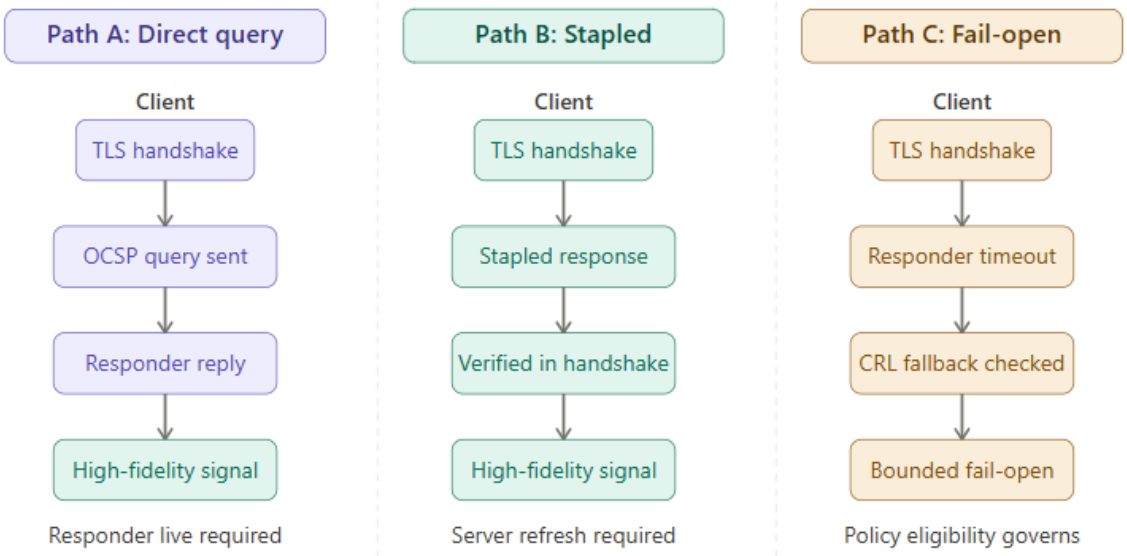


**Figure 2: Ocsp Delivery Paths At Validation Time [18, 20]**

### 3.3 Complementary Failure Characteristics and the Case for Layering

In contrast to OCSP, the architectures of CRLs cause CRL-based systems to progressively fail as the data becomes obsolete or is no longer available to the consumer. Thus, OCSP-based systems will have a fail-stop behavior whenever the responder's services are unavailable or overburdened. Neither method, however, is completely reliable under all circumstances, and both distribution and service failures are common when working at the Internet scale [5, 7]. DNSSEC is another Internet-scale cryptographic system that offers a rough comparison of these costs. The problems with DNSSEC were not so much with its cryptographic design but rather with the system's distributed nature, DNS caching, and the heterogeneous mix of crusty administrative domains that it has to deal with. Certificate revocation faces structurally similar problems [7].

Other than using CRLs and OCSP, revocation behavior is determined by how relying parties are expected to deal with a lack of knowledge about revocation status, i.e., the architectural assumption of continuity of trust under uncertainty [15]. Whether a validation fails or passes in the absence of revocation signals determines how the absence of revocation signals is read upstream in application-level trust decisions. For API gateways and security enforcement proxies, the difference is in the configured defaults for fail-open and fail-closed modes based on the criticality of the endpoint [4].

For architectural purposes, CRLs, OCSP, and fallback validation behaviors may be considered different but complementary signals of trust with different strengths and weaknesses [18]. CRLs provide durable, cacheable revocation information that is useful in disconnected or constrained environments, whereas OCSP provides higher-fidelity, time-sensitive revocation information that is optimized for more powerful, connected environments. The validation behavior under uncertain conditions influences the permissiveness of the system with respect to partial knowledge. This motivates the combination of different revocation signals in a layered approach to meet trust resilience requirements without relying on individual mechanisms [24]. Scalability and privacy limitations inherent in both CRL distribution and OCSP querying have prompted research into alternative revocation architectures. Distributed cryptographic accumulator-based schemes, for instance, have been proposed to address the bandwidth overhead of CRLs and the responder dependency of OCSP while preserving verifiable revocation semantics. The existence of such proposals reinforces the case for a layered model that is mechanism-agnostic in its upper layers, capable of accommodating new revocation signal types as they mature without requiring the confidence-ordering architecture to be restructured [27].

| Mechanism | Delivery Model | Freshness | Failure Mode | Signal Strength |
|---|---|---|---|---|
| **CRL (Full)** | Pre-distributed | Hours to days | Gradual staleness | Durable |
| **Delta CRL** | Incremental district | Hours (delta) | Delta sync failure | Durable |
| **OCSP (Direct)** | Real-time query | Near real-time | Abrupt on outage | High Fidelity |
| **OCSP Stapling** | Handshake bound | Server refresh rate | Stale staple risk | High Fidelity |
| **Fail-Open** | Policy fallback | Indeterminate | Miscalibration | Bounded Only |

**Table 1: Revocation Mechanism Comparison [14, 17, 18]**

## 4. THE LAYERED REVOCATION RESILIENCE MODEL

### 4.1 CRLs as a Baseline Trust Signal

In the layered revocation resilience model, CRLs are the key trust signals. The primary architectural values of CRLs are their durability and independence from real-time availability. CRLs are designed in such a way that they can be distributed in advance, cached locally, and reused across multiple validation events, which makes them resilient to transient network failures and connectivity issues [17]. This characteristic is highly effective in deployments when continuous access to external services is not available. It also includes the restricted outbound connectivity of government services, IoT networks, and enterprise environments [3, 16].

From a resilience perspective, CRLs provide a baseline view of revocation state that can be consulted even when other revocation mechanisms are unavailable. This baseline does not guarantee perfect freshness, but it establishes a minimum level of revocation awareness that persists across validation contexts [14]. In this respect, the layered model treats CRL-based revocation confidence as a continuous variable that diminishes with time rather than a binary condition that either holds or fails. As the current time approaches and then exceeds the CRL's nextUpdate field, confidence in the revocation data declines in a measurable and predictable way [17, 19].

Furthermore, CRLs ensure the continuation of trust decisions under imperfect conditions, as they degrade gradually and not abruptly. This behavior also allows the system to identify the reason regarding uncertainty in a standard manner [14]. However, CRLs alone are insufficient to meet the needs of modern Internet-scale PKI. Their periodic update model introduces unavoidable latency between revocation events and global visibility. As PKI deployments have expanded in scope and scale, including certificate populations in the billions across web, IoT, and identity infrastructure, the distribution overhead and freshness limitations of CRLs impose practical constraints that the baseline layer alone cannot resolve [3, 9].

### 4.2 OCSP as a High-Fidelity Revocation Layer

OCSP operates on the second layer of the revocation resilience model. When the relying party can obtain a result in real time, OCSP allows for relying parties to determine whether a certificate has been revoked close to its time

of use. This regularizes the revocation state to be more consistent with the decision to validate [18]. The layered model considers OCSP a signal to be used in combination with baseline revocation data. Where connectivity to the OCSP responder and the OCSP responder's availability permit, OCSP responses provide additional confidence by reducing the time gap between revocation and enforcement [13].

On the other hand, OCSP has a very different set of requirements than CRLs, namely low-latency access to responder infrastructure and stable responder behavior under load, which makes OCSP particularly sensitive to network partitions, service downtime, and degraded performance [15, 18]. When analyzed from the resiliency perspective, OCSP improves accuracy and reduces failure resiliency when OCSP acts as the sole revocation authority. More advanced designs for OCSP in constrained settings, such as the smart grid and energy storage systems, address the availability-freshness trade-off by aggregating responses in the network and caching responses locally [13].

This concern is reduced by the layered approach, since OCSP is treated as a conditional enhancement layer that improves evaluative trust when present but is not essential for establishing the base trust of CRLs. This allows the system to maintain the precision of OCSP while still being strong against its intrinsic weaknesses as a single point of failure and not tightly bound to real-time availability constraints [14]. By explicitly distinguishing between baseline and high-fidelity revocation layers, the model clarifies how revocation confidence can be composed from multiple signals. CRLs provide continuity and resilience, while OCSP provides freshness and precision. Neither is sufficient in isolation, but together they form a more robust foundation for trust evaluation under diverse operational conditions [17].

### 4.3 Fail-Open Behavior as a Bounded Resilience State

In the context of revocation resilience, fail-open behavior refers to guaranteeing validation even if the revocation status of the object is unknown with sufficient certainty. Preventing fail-open behavior is generally not considered the default policy or the best practice for layered revocation resilience. Its bounded architectural state acknowledges the epistemic limits of revocation certainty in adverse conditions [15, 22]. For example, in security control infrastructures like API gateways or web application firewalls (WAFs), these defaults are explicitly mapped to the criticality of the endpoint being protected. For example, idempotent and low-risk endpoints may accept fail-open scenarios while high-privilege or high-assurance endpoints require enforced fail-closed [15]. The same applies in the context of certificate validation.

In this model of layered trust, fail-open does not actually disable revocation enforcement but provides a route to a graceful failure when there are ambiguous revocation signals. When running in this layered trust model and uncertain about whether to continue with baseline or high-fidelity revocation signals, failing open may be the next-best option. Rather, they are stuck with what is known [23]. It has been proposed that bounded fail-open behavior could avoid larger outages due to transient revocation infrastructure failures and make the presence of uncertainty in the trust model more explicit. Fail-open must be defined as a state in itself (and not as an implicit fallback in case of ambiguity) to avoid mixing revocation ambiguity with validation success [15].

The fail-open layer is not intended as a replacement for or to diminish the role of CRLs or OCSP, but rather is entered only when other more certain revocation signals are unavailable. The CA/Browser Forum Baseline Requirements (BR), the Chromium Policy, and the Mozilla Policy balance revocation checking with the realistic exceptions that allow for availability and user experience [23, 24]. By bounding fail-open behavior within a layered architecture, the model enables more predictable trust outcomes under failure. Trust decisions remain anchored to explicit revocation signals whenever possible, and uncertainty is treated as a managed condition rather than an unexamined side effect of system failure [14, 22].

| Certificate Type | CRL Current OCSP Expired | CRL Stale OCSP Confirmed | CRL Unavailable OCSP Unreachable | Both Unavailable |
|---|---|---|---|---|
| **DV (Domain)** | Proceed | Proceed | Fail-open | Fail-open |
| **OV (Organization)** | Proceed | Proceed | Fail-open | Hard fail |
| **EV (Extended)** | Proceed | Escalate | Hard fail | Hard fail |
| **Client Authorization** | Proceed | Escalate | Hard fail | Hard fail |

**Table 2: Fail-Open Eligibility Decision Matrix [10, 22, 23]**

## 5. LAYER INTERACTION AND GRACEFUL DEGRADATION

### 5.1 Signal Composition and Confidence Ordering

The strength of the layered revocation resilience model lies not in the individual mechanisms it incorporates but in how those mechanisms interact under varying operational conditions. Rather than enforcing revocation through a single decision point, the model enables trust evaluation to degrade gracefully as the availability or fidelity of revocation signals diminishes [22]. When all layers are functioning as intended, trust evaluation incorporates both baseline and high-fidelity revocation signals, producing decisions with strong temporal accuracy and high confidence. Depending on the situation, e.g. on whether real-time validation is not possible, trust value evaluation may use other incoming signals, e.g. cached or periodically broadcasted signals, instead of collapsing [17, 18].

Graceful degradation takes place by ordering trust signals on the basis of confidence and availability rather than mutual exclusivity [14]. CRLs carry through validation occasions and situations, while OCSP provides more precision when possible. The order of these options is not arbitrary, with fail-open being the last choice that is only used when no other explicit revocation information is available. This ordering is to maximize the strongest available signal for any given situation. The layered model thus combines the benefits of CRLs (consistency and lower cost), OCSP (responsiveness) and stapling (performance), while providing policy-driven fail-open guarantees [21, 22].

The signal composition approach also finds precedent in large-scale cryptographic deployment experience. The DNSSEC deployment, for instance, demonstrated that layering caching, signing hierarchies, and fallback resolution behaviors was essential to maintaining availability and consistency in a distributed system subject to partial failures [7]. Similarly, Internet-scale key establishment research has shown that combining efficient symmetric operations with hierarchical key derivation produces authentication systems that degrade gracefully as individual components encounter failures [6]. These precedents reinforce the architectural logic of the layered revocation model.

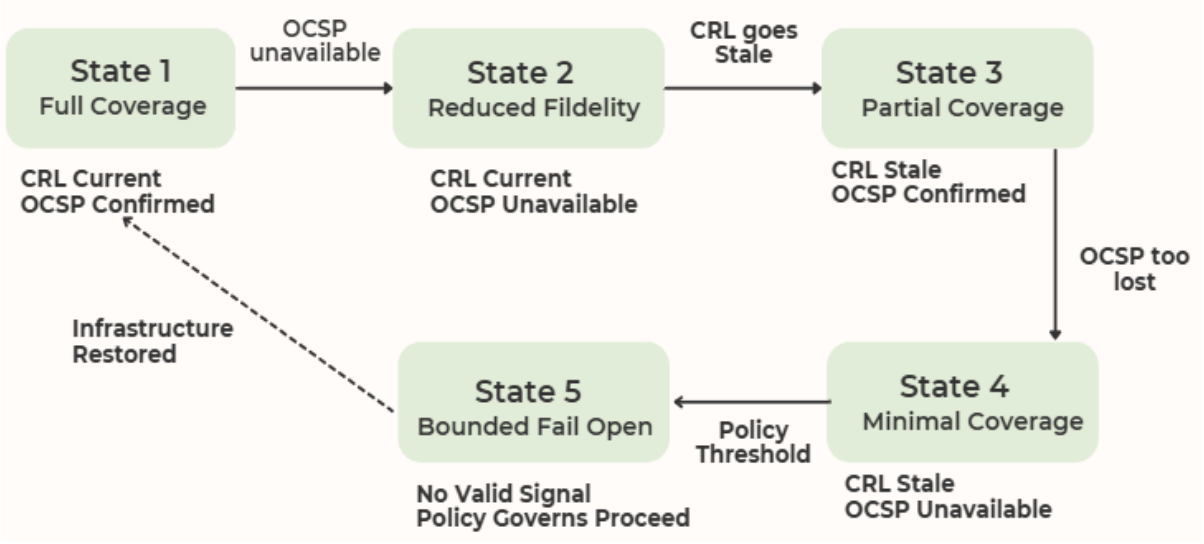


**Figure 3: Revocation Confidence Degradation Path [14, 22]**

### 5.2 Uncertainty Propagation and Predictability

Layer interaction also clarifies how revocation uncertainty propagates through systems. Rather than presenting uncertainty as an error condition that immediately halts validation, the layered model frames uncertainty as a variable that increases as higher-fidelity signals are lost [15]. This framing aligns trust evaluation with the realities of distributed systems, where partial failure is the norm rather than the exception. It is consistent with how security enforcement infrastructure handles risk under uncertainty in other contexts, where risk scores are computed from available signals and mapped to actions through calibrated policy thresholds rather than binary conditions [4, 15].

Observability is a critical enabler of this predictability. A system that silently transitions from high-confidence revocation evaluation to fail-open behavior provides no signal to operators that the underlying infrastructure has degraded [15, 22]. Observable states should surface the current confidence level of each revocation layer, the age and source of cached revocation data, the frequency and duration of OCSP query failures, and the rate at which

fail-open behavior is being invoked. In gateway and enforcement infrastructure, signed audit logs and hot-reloadable policy bundles serve exactly this purpose, providing both runtime governance and post-incident analysis capability [15]. Empirical measurement of revocation and replacement practices in production environments has found that operators frequently take hours to days to complete the revocation and re-issuance cycle following a confirmed compromise event, and that replacement certificates sometimes carry residual metadata inconsistencies that extend the effective propagation delay window [26]. These findings confirm that the gap between authoritative revocation state and relying-party visibility is a systemic operational pattern. Surfacing this gap in real time through observable layer states is therefore not merely an operational convenience but a prerequisite for managing revocation uncertainty responsibly [26]. Persistent elevation of fail-open invocation rates serves as a leading indicator of revocation infrastructure health problems, prompting corrective action before degraded conditions escalate.

By structuring revocation evaluation as a layered process with defined degradation paths, the model improves predictability in trust behavior. Systems can keep working even when conditions are bad, as long as they make it clear that those conditions are less reliable [22]. This approach does not eliminate revocation risk, but it enables systems to reason about and manage that risk more coherently across diverse operational environments, including IoT deployments with limited connectivity, government service infrastructure with high assurance requirements, and web-scale TLS deployments with heterogeneous client populations [16].

## 6. SECURITY, AVAILABILITY, AND TRUST TRADE-OFFS

### 6.1 Balancing Security Enforcement and Systemic Fragility

Certificate revocation operates at the intersection of security, availability, and trust predictability. Strengthening one dimension often puts pressure on the others, particularly in Internet-scale PKI systems where network reliability and infrastructure reachability cannot be assumed [1, 5]. From a security perspective, revocation minimizes the window during which compromised or misissued certificates remain trusted. Domain Validation (DV) supports the overwhelming majority of certificate issuance, and the security of that process depends substantially on the integrity of certificate lifecycle management, including timely revocation [10]. Attacks that exploit weaknesses in domain validation demonstrate that the consequences of revocation failure extend beyond stale status information. They can undermine the trust anchor that the entire certificate ecosystem relies upon [11]. Mechanisms with higher temporal fidelity, such as real-time OCSP queries, reduce the window during which compromised credentials can be accepted [18]. However, these mechanisms introduce dependencies on online infrastructure whose failure can abruptly disrupt validation. When availability assumptions are violated, treating security as the sole objective can therefore increase systemic fragility [15]. Research on PKI interoperability has consistently shown that legal, organizational, and technical differences across deployments make uniform enforcement difficult to sustain and that creative solutions, including middleware, trust brokers, and layered policy frameworks are needed to manage this complexity [5].

The primary security concern specific to the layered model is the risk of confidence threshold miscalibration. If the thresholds for fail-open eligibility are set too permissively, the model effectively reduces to unrestricted fail-open in practice. Threshold calibration should be treated as a security parameter, subject to review and adjustment based on observed infrastructure performance and threat model evolution [14]. This mirrors the approach taken in API gateway security, where per-endpoint thresholds with hysteresis are used to map risk space to enforcement action, ensuring that policy does not inadvertently collapse into uniform permissiveness [16].

### 6.2 Availability Continuity and Trust Predictability

The best way to support availability is to have the fewest possible dependencies on the revocation infrastructure so that a temporary failure does not cause a cascade into a full service outage [15]. Cacheable revocation base signals can support availability in scenarios such as the Internet of things (IoT) and smart homes, where devices may lack reliable and constant Internet access but may still need to perform certificate validation operations to help protect their communications. In such cases, the presence of cached CRL data as a backup mechanism is an implicit necessity [16]. Trust predictability is a different problem that is often overlooked. Where trusted services and revocation behavior are enabled by real-time infrastructure, discrepancies in trust may arise, and the emergence of autonomous AI agents in an industrial internet style magnifies the problem further [9, 23]. These agents need to make trust decisions at machine timescales. Today's certificate workflows, including revocation, are not designed for machine efficiency [9].

The layered model reduces the impact of uncertainties by ordering revocation signals and explicitly limiting them, which allows trust behavior to be more stable under less than ideal conditions [22]. The trade-offs in revocation at scale are not solely an artifact of a particular mechanism or protocol design. The layered revocation resilience model makes these trade-offs an explicit architectural concern, where it is possible to trade off security, availability, or trust continuity, rather than optimizing each independently of the others [14]. Industry best practices (CA/Browser Forum) and policy guidance (Chromium and Mozilla) operationalize this balance, providing a policy layer for governing how these mechanisms work together across the many clients the ecosystem serves [23, 24]. The transition toward post-quantum cryptography introduces an additional dimension to these trade-offs. A systematic review of post-quantum migration practices found that certificate lifecycle management infrastructure, including revocation mechanisms, is among the most operationally complex components to migrate, because changes to certificate formats and signature algorithms affect the interpretability of revocation artifacts by clients at different stages of migration maturity [29]. During hybrid migration periods, when both classical and post-quantum certificate configurations are in active deployment, the layered model's confidence-ordering architecture must accommodate the possibility that different client cohorts interpret the same revocation artifact differently, producing trust outcome divergence that neither the distribution nor the query layer can resolve independently [29].

## CONCLUSION

Certificate revocation is not merely a protocol-level concern. It is a distributed systems problem that sits at the intersection of cryptographic trust, infrastructure availability, and operational consistency. As PKI has grown to underpin web security, government digital services, IoT deployments, and large-scale identity systems, the gap between revocation state and revocation visibility has become a defining challenge for trust infrastructure architects. No single mechanism, whether CRL-based distribution, real-time OCSP querying, or stapled response delivery, is sufficient to address this challenge across the full range of operational conditions that Internet-scale deployments encounter. The layered revocation resilience model presented in this article offers a structured response to this reality. By organizing CRLs, OCSP, and bounded fail-open behavior into a confidence-ordered hierarchy, the model enables trust evaluation to degrade predictably rather than fail abruptly. Each layer contributes a distinct and non-substitutable property: durability, temporal precision, and bounded continuity under uncertainty. The architectural trade-offs among security enforcement, availability, and trust predictability are treated as first-class design considerations rather than incidental policy choices. Revocation uncertainty, in this model, is not an error condition to be suppressed. It is a managed state to be surfaced, bounded, and resolved through layered signal composition. This perspective provides a foundation for reasoning about revocation resilience in environments where ideal conditions cannot be assumed and where the consequences of trust failure are significant.

This work leaves several avenues for future work: first, the theoretical fail-open eligibility conditions and current cloud architecture gap require further empirical validation. The fail-open eligibility criteria in Table 2 are motivated by cloud architectural principles. However, more large-scale measurement studies are needed for confirmation and perhaps tuning. The relation of the layered model to other emerging revocation architectures, such as short-lived certificate schemes and distributed accumulator-based architectures, is also worth exploring. The revocation architectures above may change the freshness and distribution characteristics of revocation signals, and these changes may require updating or extending the layer ordering and degradation paths provided above. Third, the capability of the layered approach in the context of post-quantum PKI transitions may need to be investigated. There are several changes that might be necessary due to updates in certificate format, signature algorithm, and key exchange construction, which affect how legacy revocation artifacts can be understood by transitioning clients. It is also possible to implement the model in production environments with limited deployment capacity (e.g., fleets of IoT devices and energy management systems that are occasionally online). This would, however, require additional engineering to transform the architecture into deployment specifications. Finally, observability requirements discussed in Section 5.2 point to opportunities for further work on monitoring the revocation infrastructure. Standardizing metrics, alert thresholds, and audit log schema reporting layer-level confidence degradation in real time would make the model easier to operate in production.

**REFERENCES**

[1] ROZLINDA Radzali et al., "Analysis of trust models in public key infrastructure: A systematic literature review of interoperability challenges," Malaysian Journal of Science Health & Technology, 2025. Available: https://mjosht.usim.edu.my/index.php/mjosht/article/view/465

[2] Christopher Tullis et al., "Electronic Signatures: Enabling Trusted Digital Transformation," World Bank Digital Transformation White Paper Series, 2024. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5180929

[3] Oleksandr Dulia and Dmytro Minochkin, "An exploration of public key infrastructure applications across diverse domains: a comparative analysis," Collection "Information Technology and Security," 2023. Available: https://its.iszzi.kpi.ua/article/download/293496/288052

[4] Ramanan Hariharan, "API Gateway Threat Prevention in Large-Scale Applications," International Journal of Sustainability and Innovation in Engineering, 2024. Available: https://www.doi.org/10.56830/IJSIE092024

[5] Massimiliano Pala, "A proposal for collaborative Internet-scale trust infrastructures deployment: the Public Key System (PKS)," in Proceedings of the 9th Symposium on Identity and Trust on the Internet, 2010. Available: https://dl.acm.org/doi/pdf/10.1145/1750389.1750404

[6] Benjamin Rothenberger et al., "PISKES: Pragmatic Internet-scale key-establishment system," in Proceedings of the 15th ACM Asia Conference on Computer and Communications Security, 2020. Available: https://dl.acm.org/doi/pdf/10.1145/3320269.3384743

[7] Hao Yang et al., "Deploying cryptography in Internet-scale systems: A case study on DNSSEC," IEEE Transactions on Dependable and Secure Computing, 2010. Available: https://ieeexplore.ieee.org/abstract/document/5444890/

[8] Jan Rüth et al., "Large-scale scanning of TCP's initial window," in Proceedings of the 2017 Internet Measurement Conference, 2017. Available: https://dl.acm.org/doi/pdf/10.1145/3131365.3131370

[9] Ramesh Raskar et al., "Upgrade or Switch: Do We Need a Next-Gen Trusted Architecture for the Internet of AI Agents?," in 2025 IEEE 7th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (TPS-ISA), 2025. Available: https://arxiv.org/pdf/2506.12003

[10] Markus Brandt et al., "Domain validation++ for mitm-resilient pki," in Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, 2018. Available: https://dl.acm.org/doi/pdf/10.1145/3243734.3243790

[11] Pouyan Fotouhi Tehrani et al., "Security of alerting authorities in the WWW: Measuring namespaces, DNSSEC, and Web PKI," in Proceedings of the Web Conference 2021, 2021. Available: https://dl.acm.org/doi/pdf/10.1145/3442381.3450033

[12] Kunlun Xu et al., "Distribution-aware knowledge prototyping for non-exemplar lifelong person re-identification," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024. Available: https://openaccess.thecvf.com/content/CVPR2024/papers/Xu_Distribution-aware_Knowledge_Prototyping_for_Non-exemplar_Lifelong_Person_Re-identification_CVPR_2024_paper.pdf

[13] Hong-Sheng Huang et al., "An Enhanced Online Certificate Status Protocol for Public Key Infrastructure with Smart Grid and Energy Storage System," arXiv preprint arXiv:2409.10929, 2024. Available: https://arxiv.org/pdf/2409.10929

[14] Torben Pedersen, "CPS, certificate practice statement," in Encyclopedia of Cryptography, Security and Privacy, Springer Nature Switzerland, 2025. Available: https://link.springer.com/content/pdf/10.1007/978-3-030-71522-9_283.pdf

[15] Ramanan Hariharan, "API Gateway Threat Prevention in Large-Scale Applications," International Journal of Sustainability and Innovation in Engineering, 2024. Available: https://journals.scipubhouse.com/IJSIE/article/download/258/253

[16] Amitabh Mishra et al., "Trade-offs involved in the choice of cloud service configurations when building secure, scalable, and efficient Internet-of-Things networks," International Journal of Distributed Sensor Networks, 2020. Available: https://journals.sagepub.com/doi/pdf/10.1177/1550147720908199

[17] D. Cooper et al., "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile," Internet Engineering Task Force, Request for Comments 5280, 2008. Available: http://www.ietf.org/rfc/rfc5280.txt

[18] S. Santesson et al., “X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP," Internet Engineering Task Force, Request for Comments 6960, 2013. Available: http://www.ietf.org/rfc/rfc6960.txt

[19] A. Deacon and R. Hurst, “The Lightweight Online Certificate Status Protocol (OCSP) Profile for High-Volume Environments,” 2007. Available: https://www.ietf.org/rfc/rfc5019.txt

[20] D. Eastlake, “Transport Layer Security (TLS) Extensions: Extension Definitions, Internet Engineering Task Force,” Request for Comments 6066, 2011. Available: http://www.ietf.org/rfc/rfc6066.txt

[21] Eric Rescorla, “The transport layer security (TLS) protocol version 1.3. No. rfc8446,” Internet Engineering Steering Group, 2018. Available: https://www.ietf.org/rfc/rfc8446.txt

[22] CA/Browser Forum, “Baseline Requirements.” Available: https://cabforum.org/baseline-requirements/

[23] Google Chrome, “Chrome Root Program Policy, Version 1.8,” 2026. Available: https://www.chromium.org/Home/chromium-security/root-ca-policy/

[24] Mozilla, “Mozilla Root Store Policy,” 2026. Available: https://www.mozilla.org/en-US/about/governance/policies/security-group/certs/policy/

[25] Markus Sosnowski et al., "An Internet-Wide View on HTTPS Certificate Revocations: Observing the Revival of CRLs via Active TLS Scans," in Proc. 2024 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), IEEE, 2024. Available: https://www.net.in.tum.de/fileadmin/bibtex/publications/papers/sosnowski2024certificates.pdf

[26] Lalchandra Rampersaud et al., “Evaluating Security Checks Against Malicious Payloads with Forged Signatures,” in 2025 IEEE 16th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), pp. 0516-0525. IEEE, 2025. Available: https://akhavani.net/publications/browser-blindspot/browser-blindspot.pdf

[27] Munshi Rejwan Ala Muid et al., "AccuRevoke: Enhancing Certificate Revocation with Distributed Cryptographic Accumulators," in Proc. 2025 IEEE Symposium on Security and Privacy (SP), IEEE, 2025. Available: https://thanghoang.github.io/publication/25_sp_accurevoke/25_sp_accurevoke.pdf

[28] Hyunsoo Kwon et al., "Certificate Revocation in the TLS Ecosystem: A Survey," ACM Computing Surveys, 2026. Available: https://dl.acm.org/doi/pdf/10.1145/3785653

[29] Christian Näther et al., "Migrating Software Systems Toward Post-Quantum Cryptography: A Systematic Literature Review," IEEE Access, 2024. Available: https://ieeexplore.ieee.org/iel8/6287639/6514899/10648683.pdf

[30] N. Fernandes, “Multicultural competence in counselling practice: Implications for sexual health interventions,” Sarcouncil Journal of Arts, Humanities and Social Sciences, vol. 2, no. 4, pp. 28–35, 2023.

[31] V. Sahoo, “A machine learning-based framework for agile product development and growth strategy optimization,” Journal of Information Systems Engineering and Management, vol. 7, no. 3, 2022. [Online]. Available: https://jisem-journal.com/index.php/journal/vol7_iss3