ENHANCING DIGITAL TRUST WITH DNS-BASED ROOT CERTIFICATE

RE-VALIDATION

Roble Mumin

AGENDA

1. Introduction
2. Threats
3. Approach
4. Detail
5. Configuration

COMBATTING CYBERSECURITY THREATS THROUGH ROOT CERTIFICATE RE-VALIDATION

WHY?

The Backbone of Digital Security

CERTIFICATES: Essential for authentication and identification, certificates underpin secure digital communications, encryption, and access control, forming the backbone of cybersecurity infrastructure.

Vulnerabilities and Risks

RISKS: Compromised certificates, including spoofed or malicious entries, pose severe threats, breaking the security chain. They enable attackers to decrypt data, install malware, and conduct man-in-the-middle attacks under a guise of legitimacy.

Proactive Defense with Root Certificate Re-Validation

SOLUTION: The DNS-based Root Certificate Re-Validation offers an approach to mitigating these risks by periodically re-validating root certificates. This process ensures their legitimacy, restoring trust and integrity to the system's security mechanisms.

SECURING DIGITAL TRUST WITH DNS-BASED RE-VALIDATION AND DEDICATED DOMAINS

WHAT?

Mechanism

Leveraging DNS for Security: Basic Mode utilizes the DNS infrastructure, incorporating a dedicated domain (e.g., .cert/.certs) for storing and accessing hashed root certificate data. This approach enables automated, real-time validation of certificates before they are trusted for use.

The Role of the Dedicated Domain

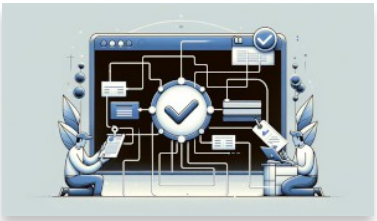
Dedicated Domain Advantage: By assigning a dedicated domain (.cert/.certs) for certificate hashes, Basic Mode creates a centralized, easily accessible repository for validation purposes. This ensures that only certificates verified against this trusted DNS repository are accepted, significantly reducing the risk of certificate spoofing or tampering.

Mitigating Vulnerabilities and Enhancing Trust

Enhancing Cybersecurity: DNS-based Root Certificate Re-Validation through the Basic Mode and its dedicated domain mechanism effectively mitigates vulnerabilities associated with compromised certificates. It prevents the use of unauthorized certificates, thereby maintaining the integrity and trustworthiness of digital communications and transactions.

CHOREOGRAPHY OF DNS-BASED CERTIFICATE RE-VALIDATION PROCESS

HOW?



Initiation of the Query Process

Initiating Certificate Validation: The process begins when a client or system needs to validate a certificate's authenticity. A DNS query is generated for the certificate's hashed data stored under the dedicated .cert/.certs domain, initiating a verification request.

Retrieving Certificate Data via DNS

DNS Response and Data Retrieval: The DNS server responds to the query with the certificate's hashed data from the .cert/.certs domain. This data includes the certificate's hash value and any relevant metadata necessary for validation, ensuring that the information is fetched in real-time and is up-to-date.

Verification and Trust Establishment

Validating and Establishing Trust: The client compares the fetched hash from the DNS response against the local hash of the certificate in question. If the hashes match, the certificate is validated as authentic and trustworthy. This process effectively mitigates risks of certificate spoofing, tampering, or other malicious interventions, reinforcing the security infrastructure.

CONFIGURATION AND NAMESPACE STRUCTURE OF THE .CERT/.CERTS DOMAIN

A LITTLE SOMETHING LIKE THIS!

TLD Domain Setup and Configuration

Domain Configuration for Security: The .cert/.certs TLD is specifically designed for storing hashed certificate data. Its configuration is optimized for security and efficiency, with strict access controls and protocols ensuring that only authorized queries can retrieve certificate hashes. DNSSEC is deployed to guarantee the integrity and authenticity of the data within this domain.

Structuring the Namespace for Certificate Data

Namespace Organization: Within the .cert/.certs domain, certificate hashes are organized using a structured naming convention. This could involve segmenting the namespace by certificate issuer, type, or validity period. For example, issuername.cert/issuername.certs could be a structure to store hashes of certificates issued by a particular authority, facilitating efficient and targeted queries.

Practical Usage and Retrieval Mechanisms

Efficient Query and Retrieval: To validate a certificate, a client constructs a DNS query using the structured namespace, targeting the specific segment relevant to the certificate in question. This query retrieves the stored hash for comparison, streamlining the validation process. Suggestions for best practices include regular updates to the DNS records to reflect changes in certificate status and the use of caching mechanisms to enhance query response times.

CONTACT INFORMATION

Name: Roble Mumin

Position: Manager, KPMG Germany, Cybersecurity in the Public Sector

Email: ietf@roblemumin.com

LinkedIn: https://www.linkedin.com/in/roblemumin