​

**Scenario: Secure Storage and Retrieval of a Phone Number**

In this scenario, a phone number is classified as **highly sensitive** and must be encrypted before being stored in a database and securely decrypted when retrieved by an authorized user.

**Step-by-Step Process in D-CAF**

**Step 1: Sensitivity Classification**

The phone number is classified as **highly sensitive** because it contains personal identifiable information (PII) that must be protected due to privacy regulations such as GDPR or HIPAA.

* **D-CAF Action:** The system assigns a **sensitivity score** (since the phone number is highly sensitive).

**Step 2: Context and Risk Assessment (Pre-Encryption)**

Before encryption, D-CAF evaluates the context in which the phone number is being handled and the risks associated with storing or transmitting this data.

1. **Source Risk (R\_s):**
   * The phone number originates from a secure internal application, so the **source risk** is assessed as low
2. **Destination Risk (R\_d):**
   * The phone number will be stored in an encrypted database in a secure data center, so the **destination risk** is low
3. **Network Condition Risk (R\_n):**
   * The network is a secure internal network using TLS, so the **network condition risk** is low
4. **Contextual Risk:**
   * **Geographic Location (C\_l):** The data is being handled within a trusted geographic region,
   * **Time of Access (C\_t):** Access is during regular business hours, so time risk

**Step 3: Composite Risk Score Calculation**

D-CAF calculates the total risk score combining the flow risks and the contextual risks:

Contextual risk is calculated as:

Ce=

Finally, the **total risk score**:

**Step 4: Encryption Strength Calculation**

Now, D-CAF calculates the required **encryption strength** ​ based on the sensitivity of the phone number and the total risk score.

Based on this value, we refer to the encryption strength thresholds defined by D-CAF:

Since , D-CAF selects **RSA-2048** for encryption because the value falls below 2, and RSA offers more security for specific use cases in certain contexts.

**Algorithm: Dynamic Key Length Selection Based on Encryption Strength**

**Input:**

* **Encryption Strength** Es (calculated by the D-CAF algorithm)
* **Type of Encryption** (symmetric or asymmetric)

**Output:**

* Optimal key length for the encryption

**Step 1: Define Key Length Thresholds**

We first define thresholds for different ranges of Es and assign key lengths accordingly. The thresholds will vary depending on whether symmetric or asymmetric encryption is used.

**Symmetric Encryption (AES) Thresholds:**

* If use **AES-256** (256-bit key)

**Asymmetric Encryption (RSA) Thresholds:**