

GABRIELE ROMANATO

Calcolo del checksum MD5 di un file con Java

Il checksum MD5 è un hash a 128 bit rappresentato da una stringa di 32 caratteri, spesso usato per verificare l'integrità di un file. Quando si scarica un file da internet o si lavora con grandi quantità di dati, il checksum MD5 aiuta a rilevare eventuali modifiche accidentali o dannose, confrontando l'hash del file scaricato con l'hash originale. In questo articolo, vedremo come calcolare il checksum MD5 di un file in Java. Per farlo, utilizzeremo la classe `MessageDigest` fornita dalla libreria standard Java.

Java include tutto il necessario per calcolare l'MD5, quindi non è necessario aggiungere alcuna libreria esterna. Dovremo solo importare le classi `MessageDigest`, `FileInputStream` e `DigestInputStream`.

```
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;

public class MD5Checksum {

    public static String getMD5Checksum(File file) throws
IOException, NoSuchAlgorithmException {
        // Creiamo un'istanza di MessageDigest per MD5
        MessageDigest md = MessageDigest.getInstance("MD5");

        // Creiamo uno stream per leggere il file
        try (FileInputStream fis = new FileInputStream(file))
        {
            byte[] buffer = new byte[1024];
            int bytesRead;

            // Leggiamo il file a blocchi e aggiorniamo il
digest
```

```

        while ((bytesRead = fis.read(buffer)) != -1) {
            md.update(buffer, 0, bytesRead);
        }
    }

    // Convertiamo il risultato dell'hash in una stringa
    esadecimale
    byte[] md5Bytes = md.digest();
    StringBuilder sb = new StringBuilder();

    for (byte b : md5Bytes) {
        sb.append(String.format("%02x", b));
    }

    return sb.toString();
}

public static void main(String[] args) {
    try {
        File file = new File("path/del/tuo/file.txt");
        String checksum = getMD5Checksum(file);
        System.out.println("Checksum MD5: " + checksum);
    } catch (IOException | NoSuchAlgorithmException e) {
        System.err.println("Errore nel calcolo del
checksum: " + e.getMessage());
    }
}
}

```

Spiegazione del codice:

1. Creazione dell'istanza MessageDigest:

`MessageDigest.getInstance("MD5")` crea un oggetto `MessageDigest` configurato per utilizzare l'algoritmo MD5.

2. Lettura del file: Usando `FileInputStream`, leggiamo il contenuto del file in blocchi di 1024 byte. Questo è utile per gestire file di grandi dimensioni senza sovraccaricare la memoria.

3. **Aggiornamento del digest:** Il metodo `md.update(buffer, 0, bytesRead)` aggiorna l'hash con il contenuto del buffer.
4. **Calcolo del checksum:** Dopo aver letto il file, chiamiamo `md.digest()` per ottenere il checksum sotto forma di array di byte.
5. **Conversione in stringa esadecimale:** Poiché i byte non sono direttamente leggibili, li convertiamo in una stringa esadecimale. Questo viene fatto con `String.format("%02x", b)`, che converte ogni byte in una rappresentazione a due cifre in esadecimale.

Conclusione

Il calcolo del checksum MD5 in Java è semplice e permette di garantire l'integrità dei file. Ricorda che MD5, essendo un hash a 128 bit, è vulnerabile a collisioni e non è considerato sicuro per applicazioni critiche. Tuttavia, per la verifica dell'integrità di file non sensibili, MD5 è ancora molto utilizzato.