Node.js: verificare la validità del certificato SSL di un dominio

In questo tutorial vedremo come verificare la validità del certificato SSL di un sito usando Node.js.

Si tratta di effettuare una richiesta HTTPS di tipo HEAD e comparare la data di scadenza del certificato ottenuta dall'oggetto `response` e l'avvenuto stato della transazione SSL (handshake) così come riportato dalla socket sottostante.

```
'use strict';

const https = require('https');
const validator = require('validator');


const getDaysBetween = (validFrom, validTo) => {
    return Math.round(Math.abs(+validFrom - +validTo)
/ 8.64e7);
};

const getDaysRemaining = (validFrom, validTo) => {
    const daysRemaining = getDaysBetween(validFrom,
validTo);
    if (new Date(validTo).getTime() < new
Date().getTime()) {
        return -daysRemaining;
    }
    return daysRemaining;
};
```

```
const getSSLCertificateInfo = host => {
    if(!validator.isFQDN(host)) {
        return Promise.reject(new Error('Invalid
host.'));
    }
    const options = {
        agent: false,
        method: 'HEAD',
        port: 443,
        rejectUnauthorized: false,
        hostname: host
    };

    return new Promise((resolve, reject) => {
        try {
            const req = https.request(options, res =>
{
                const crt =
res.connection.getPeerCertificate(),
                    vFrom = crt.valid_from, vTo =
crt.valid_to;
                var validTo = new Date(vTo);
                resolve({
                    daysRemaining:
getDaysRemaining(new Date(), validTo),
                    valid: res.socket.authorized ||
false,
                    validFrom: new
Date(vFrom).toISOString(),
                    validTo: validTo.toISOString()
                });
            });
            req.on('error', reject);
```

```
            req.end();
        } catch (e) {
            reject(e);
        }
    });
};
```

Le proprietà che più ci interessano dell'oggetto restituito dalla Promise sono `daysRemaining` e `valid`. Possiamo usarle nel modo seguente:

```
const checkCertificateValidity = async host => {
            let isValid = true;
        try {
            const res = await
getSSLCertificateInfo(this.host);
            if(res.daysRemaining <= 0 || !res.valid)
{
                isValid = false;
            }
        } catch(err)  {
            isValid = false;
        }

        return isValid;
};
```

Questa soluzione ci permette di evitare di dover usare utility SSL dalla shell che ci costringerebbero a passare un parametro arbitrario ai comandi.