

Creare uno script Bash per verificare la validità del certificato SSL di un dominio

La sicurezza online è una priorità fondamentale, e una parte essenziale di questo aspetto è rappresentata dai certificati SSL (Secure Socket Layer). Un certificato SSL crittografa la connessione tra il server e il browser, assicurando che i dati trasmessi siano protetti da accessi non autorizzati. In questo articolo, spiegheremo come creare uno script Bash che, dato un dominio di input, verifichi la validità del suo certificato SSL.

Prima di iniziare, assicuriamoci di avere alcuni strumenti essenziali installati nel nostro sistema come OpenSSL, uno strumento open source per implementare i protocolli SSL/TLS.

```
sudo apt-get install openssl    # Per sistemi basati  
su Debian/Ubuntu
```

Creiamo uno script Bash chiamato `check_ssl.sh` utilizzando un editor di testo come `nano` o `vim`.

```
nano check_ssl.sh
```

All'interno dello script, inseriamo il seguente codice:

```
#!/bin/bash

# Verifica se è stato fornito un dominio come
argomento
if [ $# -eq 0 ]; then
    echo "Usage: $0 "
    exit 1
fi

domain=$1

# Eseguire il comando OpenSSL per ottenere le
informazioni sul certificato
cert_info=$(openssl s_client -connect ${domain}:443 -
servername ${domain} -showcerts /dev/null | openssl
x509 -text)

# Verificare la data di scadenza del certificato
expiration_date=$(echo "${cert_info}" | grep "After"
| awk -F' : ' '{print $2}')

# Convertire la data di scadenza in formato timestamp
su sistemi Linux
expiration_timestamp=$(date -d "${expiration_date}"
+%s 2>/dev/null)

# Se la conversione non funziona, tenta un'altra
sintassi su sistemi BSD (come MacOS)
if [ -z "${expiration_timestamp}" ]; then
    expiration_timestamp=$(date -jF "%b %e %T %Y %Z"
"${expiration_date}" +%s 2>/dev/null)
fi
```

```
# Ottenere la data e l'ora correnti in formato
timestamp
current_timestamp=$(date +%s)

# Verificare se il certificato è scaduto
if [ ${current_timestamp} -gt ${expiration_timestamp}
]; then
    echo "Il certificato SSL per il dominio ${domain}
è scaduto."
else
    echo "Il certificato SSL per il dominio ${domain}
è valido fino al ${expiration_date}."
fi
```

Salviamo e chiudiamo il file. Rendiamo lo script eseguibile con il seguente comando:

```
chmod +x check_ssl.sh
```

Ora possiamo utilizzare lo script fornendo un dominio come argomento. Ad esempio:

```
./check_ssl.sh example.com
```

Lo script eseguirà una connessione SSL al dominio specificato e verificherà la data di scadenza del certificato. Restituirà un messaggio indicante se il certificato è valido o se è scaduto.

Questo script può essere utile per monitorare la validità dei certificati SSL su diversi domini e automatizzare il processo di verifica. Assicurarsi di utilizzare questo script in modo etico e rispettare le normative sulla privacy e la sicurezza dei dati.