

Esegui chiamate API XML manuali a CIMC

Sommario

[Introduzione](#)

[Componenti usati](#)

[Premesse](#)

[Esegui chiamate API XML a CIMC](#)

[Passaggio 1](#)

[Passaggio 2](#)

[Passaggio 3](#)

[Passaggio 4](#)

[Risoluzione dei problemi](#)

Introduzione

In questo documento viene descritto come eseguire chiamate API XML manuali a Cisco Integrated Management Controller (CIMC).

Componenti usati

- Una macchina Linux (qualsiasi distribuzione).
- Connettività di rete tra il computer Linux e Cisco CIMC (è sufficiente un test ping riuscito).



Nota: I nomi di file (`main.sh`, `login.xml`, `get_summary.xml`) utilizzati in questa guida sono arbitrari. È possibile utilizzare le proprie convenzioni di denominazione, a condizione che vi si faccia riferimento in modo appropriato durante l'intero processo.

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

Premesse

In questa dimostrazione, uno script Bash denominato "`main.sh`" viene utilizzato per eseguire chiamate API al server.

Si noti che anche metodi alternativi, come CURL (direttamente dalla CLI) o Python, possono essere usati per raggiungere lo stesso risultato.

Esegui chiamate API XML a CIMC

L'API XML CIMC richiede un passaggio di autenticazione iniziale tramite una chiamata API "aaaLogin". Durante questo processo, viene recuperato un cookie di sessione, che viene quindi utilizzato per autenticare tutte le chiamate API successive.

Passaggio 1

Creare innanzitutto un file denominato main.sh nel computer Linux remoto. Questo file contiene uno script bash utilizzato per inviare le chiamate API.

Lo script main.sh viene eseguito più volte in questo esercizio: prima per l'autenticazione e poi per il recupero di informazioni dal CIMC nelle chiamate successive.

Ecco il contenuto del file main.sh:

```
#!/bin/bash
# CONFIGURATION - Edit these variables
CIMC_IP="X.X.X.X" # <<<<<<< Customer CIMC IP address
USERNAME="admin"
PASSWORD="xxxxxxx" # <<<<<<< CIMC Password in plaintext
XML_PAYLOAD_FILE="login.xml" # <<<<<<< Referencing the login.xml file for authentication and cookie ret
CIMC_URL="https://${CIMC_IP}/nuova" # <<<<<<< Call is made to this URL

# Check if payload file exists
if [ ! -f "$XML_PAYLOAD_FILE" ]; then
echo "Error: XML payload file '$XML_PAYLOAD_FILE' not found."
exit 1
fi
# Send XML request using curl
curl -k -s -u "${USERNAME}:${PASSWORD}" -H "Content-Type: text/xml" -d @"${XML_PAYLOAD_FILE}" "${CIMC_U
```

Lo script bash definisce i parametri di accesso CIMC ed esegue inoltre una chiamata API XML, specificata in un altro file denominato login.xml identificato da una variabile denominata XML_PAYLOAD_FILE.

Lo script verifica inoltre se il file login.xml, definito dalla variabile XML_PAYLOAD_FILE esiste ed è un file normale.

Se il file definito da XML_PAYLOAD_FILE non esiste, lo script visualizza un errore e si chiude.

Salvare il file e renderlo eseguibile eseguendo questo comando nella CLI:

```
$ sudo chmod +x main.sh
```

Passaggio 2

Creare quindi il file `calledlogin.xml` nella stessa directory linux del file `main.sh`.

Questo file di accesso contiene la chiamata API `actualaaLoginXML` inviata al CIMC per recuperare un cookie di sessione. Il cookie recuperato viene utilizzato per le chiamate API successive:

Ricordarsi di sostituire il nome utente e la password CIMC con le credenziali appropriate.

Eseguire lo script `main.sh` nella CLI per recuperare un cookie:

```
$ sudo ./main.sh
```

Se la chiamata API ha esito positivo, la risposta XML restituita contiene una chiave `calledoutCookie` il cui valore corrisponde al cookie recuperato, come mostrato di seguito:

```
<?xml version="1.0"?>
<aaLogin cookie="" response="yes" outCookie="xxxxxxxx/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx" outRefreshPeriod="600" outPriv="admin" outSessionId="18"
outVersion="4.3(5.250001)"> </aaLogin>
```

Dall'output, individuare il valore `outCookie`.

Salva il valore del cookie.

Passaggio 3

Creare un nuovo file nella stessa directory in cui si trova il file `main.sh`. Assegnare al file il nome `get_summary.xml`.

Per un elenco completo delle richieste API che possono essere utilizzate con Cisco IMC, fare riferimento al [Cisco UCS Rack-Mount Servers CIMC XML API Programmer's Guide](#).

Il nuovo file `get_summary.xml` viene utilizzato per recuperare le informazioni di riepilogo sul server e lo stato di alimentazione dell'host. utilizzando il blocco di codice XML nella documentazione di riferimento, ma sostituendo il valore della chiave del cookie con il cookie di recupero precedente.

Sostituire <cookie_value> con il valore outCookievalue ottenuto dal precedente login.xmlresponse. L'aspetto della richiesta aggiornata è il seguente:

```
1 | <configResolveClass cookie="xxxxxxxx/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx"
    | inHierarchical="false" classId="computeRackUnit"/>
```

Sostituire <cookie_value> con il valore effettivo del cookie recuperato durante il processo di autenticazione.

Passaggio 4

Dopo aver aggiunto il cookie al nuovo file get_summary.xml, aggiornare lo script main.sh per fare riferimento a "get_summary.xml" come valore per la variabile XML_PAYLOAD_FILE per questa richiesta.

```
#!/bin/bash
# CONFIGURATION - Edit these variables
CIMC_IP="X.X.X.X"
USERNAME="admin"
PASSWORD="xxxxxxx"
XML_PAYLOAD_FILE="get_summary.xml" # <<<<<<< Referencing the get_summary.xml file for subsequent API call
CIMC_URL="https://${CIMC_IP}/nuova"
# Check if payload file exists
if [ ! -f "$XML_PAYLOAD_FILE" ]; then
echo "Error: XML payload file '$XML_PAYLOAD_FILE' not found."
exit 1
fi
# Send XML request using curl
curl -k -s -u "${USERNAME}:${PASSWORD}" -H "Content-Type: text/xml" -d @"${XML_PAYLOAD_FILE}" "${CIMC_URL}"
```

Eseguire di nuovo lo script main.sh per eseguire la richiesta API aggiornata.

```
$ sudo ./main.sh
```

Si riceve quindi una risposta API in formato XML che restituisce l'oggetto richiesto da CIMC.

```

<?xml version="1.0"?>
<configResolveClass cookie="xxxxxxxx/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx" response="yes" classId="computeRackUnit">
<outConfigs>
<computeRackUnit dn="sys/rack-unit-1" adminPower="policy" availableMemory="524288" model="UCSC-C220-M6S" memorySpeed="3200" name="UCS C220
M6S" numOfAdaptors="2" numOfCores="56" numOfCoresEnabled="56" numOfCpus="2" numOfEthHostIfs="0" numOfFcHostIfs="0" numOfThreads="112" operPower
="off" originalUuid="XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX" presence="equipped" serverId="1"
serial="XXXXXXXXXX" totalMemory="524288" usrLbl="" uuid="XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX" vendor="Cisco Systems
Inc" cimcResetReason="graceful-reboot
" assetTag="Unknown" adaptorSecureUpdate="Enabled" resetComponents="components" storageResetStatus="NA" vicResetStatus="NA" bmcResetStatus="NA" sma
rtUsbAccess="disabled" smartUsbStatus="Disabled" biosPostState="pending"/>
</outConfigs>
</configResolveClass>

```

Risoluzione dei problemi

È importante notare che la chiamata API XML aaaLogin restituisce un cookie di sessione con outRefreshPeriod di circa 600 secondi. Ciò significa che il cookie scade dopo dieci (10) minuti se non viene aggiornato ed è necessario un nuovo cookie per continuare a eseguire le richieste API.

Se si tenta di utilizzare un cookie scaduto (dopo 600 secondi), viene restituito un blocco di risposta di errore XML 552 "Autorizzazione richiesta":

```

<?xml version="1.0"?>
<configResolveDn cookie="xxxxxxxx/xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx" response="yes" errorCode="552" invocationResult="service-
unavailable" errorDescr="Authorization required"> </configResolveDn>

```

Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l'accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).