# Cisco UCS Director VIX Script Guide

## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contents</td>
<td>1</td>
</tr>
<tr>
<td>VIX Scripts</td>
<td>2</td>
</tr>
<tr>
<td>Prerequisites</td>
<td>2</td>
</tr>
<tr>
<td>Library Tasks</td>
<td>2</td>
</tr>
<tr>
<td>Execute VIX Script</td>
<td>2</td>
</tr>
<tr>
<td>Limitations</td>
<td>2</td>
</tr>
<tr>
<td>Escape Characters</td>
<td>2</td>
</tr>
<tr>
<td>Linux VM</td>
<td>4</td>
</tr>
<tr>
<td>Windows VM</td>
<td>6</td>
</tr>
<tr>
<td>RestAPIs</td>
<td>7</td>
</tr>
<tr>
<td>userAPIExecuteVIXScript (JSON based)</td>
<td>7</td>
</tr>
<tr>
<td>Limitations</td>
<td>7</td>
</tr>
<tr>
<td>Linux VM</td>
<td>8</td>
</tr>
<tr>
<td>Windows VM</td>
<td>9</td>
</tr>
<tr>
<td>userAPIExecuteVIXScriptWithOutput (JSON based)</td>
<td>11</td>
</tr>
<tr>
<td>Linux VM</td>
<td>11</td>
</tr>
<tr>
<td>Windows VM</td>
<td>13</td>
</tr>
<tr>
<td>EXECUTE_VIX_SCRIPT (XML based)</td>
<td>17</td>
</tr>
<tr>
<td>Limitations</td>
<td>17</td>
</tr>
<tr>
<td>Linux VM</td>
<td>17</td>
</tr>
<tr>
<td>Windows VM</td>
<td>18</td>
</tr>
</tbody>
</table>
VIX Scripts

A VIX script executes Linux and Windows commands using a REST API or a task in the Cisco UCS Director Task Library. Cisco UCS Director provides three APIs (JSON and XML based APIs and a library task) to send the commands to the VM on the vCenter server.

On successful execution of the commands, the VIX script shows the status code as 0. On unsuccessful execution or passing of irrelevant commands, the VIX script shows a status code other than 0.

Prerequisites

- Respective VM (Linux or Windows) must be powered on.
- Username and Password must be known for the VM.
- VMware tools must be installed.

Library Tasks

Execute VIX Script

The **Execute VIX Script** task is used to execute the Linux or Windows command on a Guest OS.

Limitations

- Command output or error message is not displayed.
- You must have the correct username and password.
- The VM must be powered on.

Escape Characters

A character preceded by a backslash (\) is an escape sequence and has special meaning to the compiler. The following table shows the Java escape sequences.

<table>
<thead>
<tr>
<th>Escape Sequence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\t</td>
<td>Insert a tab in the text at this point.</td>
</tr>
<tr>
<td>\b</td>
<td>Insert a backspace in the text at this point.</td>
</tr>
<tr>
<td>\n</td>
<td>Insert a newline in the text at this point.</td>
</tr>
<tr>
<td>\r</td>
<td>Insert a carriage return in the text at this point.</td>
</tr>
<tr>
<td>\f</td>
<td>Insert a form feed in the text at this point.</td>
</tr>
<tr>
<td>\’</td>
<td>Insert a single quote character in the text at this point.</td>
</tr>
<tr>
<td></td>
<td>Insert a double quote character in the text at this point.</td>
</tr>
<tr>
<td>---</td>
<td>-----------------------------------------------------------</td>
</tr>
<tr>
<td>\</td>
<td>Insert a backslash character in the text at this point.</td>
</tr>
</tbody>
</table>

When an escape sequence is encountered in a command, the compiler interprets it accordingly. For instance, if you want to put quotes within quotes you must use the escape sequence (\") on the interior quotes.

To print:

```
SQL instance name is "ucsd_instance1"
```

Enter the command as:

```
echo "SQL instance name is \"ucsd_instance1\" "
```

If the command contains a backslash (\), add one more \ character as prefix to the \ character to send the correct command to the VM. If the command includes a UCSD variable (${variable}), use four backslashes (\\\) instead of two.

**Examples**

```
C:\\WINDOWS\\system32\\cmd.exe mkdir C:\\test1 // two backslashes.
```

```
C:\\sqlinstall\\install.cmd ${SQL_INSTANCE_NAME} // four backslashes, since UCSD variables // are used in the command.
```

**Sample Status Codes**

- 0 – Successful execution
- 1 – ERROR_STATUS_MESSAGE (on passing invalid username and password)
- 127 – Exception thrown – Command not found
- 9009 – Exception thrown – Not recognized as an operable program or command

**Examples**

For a Linux VM, verify the status code by entering the command `echo $?`

For a Windows VM, verify the status code by entering `echo %ERRORLEVEL%`
Linux VM

Use case 1: Sending a valid command

Edit Task (Execute VIX Script)

Provide the values for the task inputs which are not mapped to workflow inputs.

- Select VM: SDK-VIX-Dont-Delete_Clon
- Credential Type: Login
- Login: root
- Password: ***********
- Script: /bin/mkdir /tmp/test123

Workflow Status

Service Request ID: 79

Apr 12, 2015 21:32:16 UTC Request submitted
Apr 12, 2015 21:32:17 UTC Executing workflow item number 1
Apr 12, 2015 21:32:17 UTC Completed workflow item number 1, with status Completed
Apr 12, 2015 21:32:23 UTC Executing workflow item number 2
Apr 12, 2015 21:32:23 UTC Trigger context executeWorkflowStep called
Apr 12, 2015 21:32:23 UTC Executing custom action Execute VIX (Execute VIX Script)
Apr 12, 2015 21:32:23 UTC Executing custom action execute VIX (execute VIX Script)
Apr 12, 2015 21:32:26 UTC Task #1 (Execute VIX (Execute VIX Script)) completed successfully in 3 seconds
Apr 12, 2015 21:32:26 UTC Input/Output values for Task #1 (Execute VIX (Execute VIX Script)): [local input: Select VM = 279]
Apr 12, 2015 21:32:26 UTC [Local Input: Credential Type = Login]
Apr 12, 2015 21:32:26 UTC [Local Input: Login = root]
Apr 12, 2015 21:32:26 UTC [Local Input: Script = /bin/mkdir /tmp/test123]
Apr 12, 2015 21:32:26 UTC [Local Input: Password = ***********]
Apr 12, 2015 21:32:26 UTC [Local Input: Direction = Output]
Apr 12, 2015 21:32:26 UTC [Local Input: EmbeddedValue = ]
Apr 12, 2015 21:32:26 UTC [Local Input: Status = ERROR]
Apr 12, 2015 21:32:26 UTC [Local Input: STATUS_CODE = 0]
Apr 12, 2015 21:32:26 UTC [Local Input: ERROR_STATUS_MESSAGE = ]
Apr 12, 2015 21:32:26 UTC Completed workflow item number 2, with status Completed
Apr 12, 2015 21:32:32 UTC Executing workflow item number 3
Apr 12, 2015 21:32:32 UTC Completed workflow item number 3, with status Completed
Use case 2: Sending an invalid command

```
CentOS release 5.4 (Final)
Kernel 2.6.10-164.el5 on an x86_64
SDK-VIX-Dont-DeleteClone login: root
Password:
Last login: Fri Apr 10 01:10:55 on tty1
[root@SDK-VIX-Dont-DeleteClone ~]# cd /tmp
[root@SDK-VIX-Dont-DeleteClone tmp]# ls
ls: /bin/ls: No such file or directory
[root@SDK-VIX-Dont-DeleteClone tmp]# echo $?
2
```

```
Workflow Status Log Objects Created and Modified Input/Output

Service Request ID: 80

Apr 12, 2015 21:39:20 UTC Request submitted
Apr 12, 2015 21:39:22 UTC Executing workflow item number 1
Apr 12, 2015 21:39:22 UTC Completed workflow item number 1, with status Completed
Apr 12, 2015 21:39:25 UTC Executing workflow item number 2
Apr 12, 2015 21:39:25 UTC Trigger context executeWorkflowStep called
Apr 12, 2015 21:39:25 UTC Executing custom action Execute VIX (Execute VIX Script)
Apr 12, 2015 21:39:25 UTC Executing custom action Execute VIX (Execute VIX Script)
Apr 12, 2015 21:39:29 UTC Task #1 (Execute VIX (Execute VIX Script)) completed successfully in 3 seconds
Apr 12, 2015 21:39:29 UTC Input/Output values for Task #1 (Execute VIX (Execute VIX Script)):
Apr 12, 2015 21:39:29 UTC [Local Input: Select VM = 279]
Apr 12, 2015 21:39:29 UTC [Local Input: Credential Type = Login]
Apr 12, 2015 21:39:29 UTC [Local Input: Login = root]
Apr 12, 2015 21:39:29 UTC [Local Input: Password = "**masked-value**"]
Apr 12, 2015 21:39:29 UTC [Local Input: Command = /bin/ls]
Apr 12, 2015 21:39:29 UTC Completed workflow item number 2, with status Completed
```
Windows VM

Use case 1: Sending a valid command

![Image of Cisco UCS Director VIX Script Guide]

Workflow Status Log Objects Created and Modified Input/Output

Service Request ID: 123

Aug 04, 2015 21:04:36 UTC Request submitted
Aug 04, 2015 21:04:40 UTC Executing workflow item number 1
Aug 04, 2015 21:04:40 UTC Completed workflow item number 1, with status Completed
Aug 04, 2015 21:04:43 UTC Executing workflow item number 2
Aug 04, 2015 21:04:43 UTC Trigger context executeWorkflowStep called
Aug 04, 2015 21:04:43 UTC Executing custom action Test (Execute VIX Script)
Aug 04, 2015 21:04:43 UTC Executing custom action Test (Execute VIX Script)

Aug 04, 2015 21:04:47 UTC Task #1 [Test (Execute VIX Script)] completed successfully in 3 seconds
Aug 04, 2015 21:04:47 UTC Input/Output values for Task #1 (Test (Execute VIX Script)):
Aug 04, 2015 21:04:47 UTC [Local Input: Select VM = 642]
Aug 04, 2015 21:04:47 UTC [Local Input: Credential Type = Login]
Aug 04, 2015 21:04:47 UTC [Local Input: Login = administrator]
Aug 04, 2015 21:04:47 UTC [Local Input: Password = **masked-value**]
Aug 04, 2015 21:04:47 UTC [Local Input: Script = C:\WINDOWS\system32\cmd.exe mkdir C:\\Test]
Aug 04, 2015 21:04:47 UTC [Local Input: Display = true]

Aug 04, 2015 21:04:47 UTC Completed workflow item number 2, with status Completed
Aug 04, 2015 21:04:49 UTC Executing workflow item number 3
Aug 04, 2015 21:04:49 UTC Completed workflow item number 3, with status Completed
Use case 2: Sending an invalid command

RestAPIs

userAPIExecuteVIXScript (JSON based)

Limitations

- The response displays only the status code. The response does not display the output of the valid command.
To view error messages, use the `userAPIExecuteVIXScriptWithOutput` command.

**Linux VM**

**Use case 1: Sending a valid command**

```
/app/api/rest?formatType=json&opName=genericvm:userAPIExecuteVIXScript&opData=
{param0:279,param1:"root",param2:"doupl123",param3:"/bin/mkdir /tmp/test123"}
```

![Execute API](image)

**Response**

```
{"serviceResult":1,"serviceError":null,"serviceName":"InfraMgr",
"opName":"genericvm:userAPIExecuteVIXScript"
}
```

```
Apr 12, 2015 21:32:26 UTC [Local Input: Credential Type = Login]
Apr 12, 2015 21:32:26 UTC [Local Input: Login = root]
Apr 12, 2015 21:32:26 UTC [Local Input: Password = **masked value**]
Apr 12, 2015 21:32:26 UTC [Local Input: Script = /bin/mkdir /tmp/test123]
Apr 12, 2015 21:32:26 UTC [Output: EXIT STATUS_CODE = 0]
```
Use case 2: Sending an invalid command

API Access
HTTP Method: GET
Method: userAPIExecuteVIXScript
Sample URL
/app/api/rest?formatType=json&opName=genericvm:userAPIExecuteVIXScript&opData=(param0:279,param1:"root",param2:"cloudia123",param3:"/\abc\def/123")

Workflow Status

Service Request ID: 486

Apr 16, 2015 01:43:19 UTC Request submitted
Apr 16, 2015 01:43:19 UTC Executing workflow item number 1
Apr 16, 2015 01:43:19 UTC Completed workflow item number 1, with status Completed
Apr 16, 2015 01:43:22 UTC Executing workflow item number 2
Apr 16, 2015 01:43:22 UTC Trigger context executeWorkflowStep called
Apr 16, 2015 01:43:23 UTC Executing custom action VIX Script (Execute VIX Script)
Apr 16, 2015 01:43:23 UTC Executing custom action VIX Script (Execute VIX Script)
Apr 16, 2015 01:43:26 UTC [VIXActionHandler] - account details VMName svm-Group1-SR61, Host:172.25.110.77, OS Type: red hat enterprise linux 4 (32-bit), ExitCode:1
Apr 16, 2015 01:43:26 UTC Task #1 (VIX Script (Execute VIX Script)) completed successfully in 3 seconds
Apr 16, 2015 01:43:26 UTC Input/Output values for Task #1 (VIX Script (Execute VIX Script)):
Apr 16, 2015 01:43:26 UTC [Local Input: Select VM = 279]
Apr 16, 2015 01:43:26 UTC [Local Inputs: Credential Type = Login]
Apr 16, 2015 01:43:26 UTC [Local Inputs: Login = root]
Apr 16, 2015 01:43:26 UTC [Local Inputs: Password = "masked-value"]
Apr 16, 2015 01:43:26 UTC [Local Input Script = "\abc\test\123"]
Apr 16, 2015 01:43:26 UTC [Local Input Script = "\abc\test\123"]
Apr 16, 2015 01:43:26 UTC [Debug: EXIT_STATUS_CODE = 1]
Apr 16, 2015 01:43:26 UTC [Debug: ERROR_STATUS_MESSAGE = ]
Apr 16, 2015 01:43:26 UTC Completed workflow item number 2, with status Completed
Apr 16, 2015 01:43:29 UTC Executing workflow item number 3
Apr 16, 2015 01:43:29 UTC Executing workflow item number 3, with status Completed

Windows VM

Use case 1: Sending a valid command

API Access
HTTP Method: GET
Method: userAPIExecuteVIXScript
Sample URL
/app/api/rest?formatType=json&opName=genericvm:userAPIExecuteVIXScript&opData=(param0:642,param1:"administrator",param2:"cloudia123",param3:"C:\\Windows\system32\cmd.exe\?
\dir a:\test")
Use case 2: Sending an invalid command

```
/app/api/rest7?formatType=json&opName=genericvm:userAPI:ExecuteVIXScript&opData="\{param0:642,param1:"administrator",param2:"c1oupie123",param3:"C:\WINDOWS\system32\cmd.exe abcd"\}"
```
Note: Use four backslashes in this API command if the command contains '\'. Verify the script in the service request (SR) logs.

**userAPIExecuteVIXScriptWithOutput (JSON based)**

Use this API to view the error messages of a failed command.

**Linux VM**

**Use case 1: Sending a valid command**
Request

//app/api/rest?formatType=json&opName=genericvm:userAPIExecuteVIXScriptWithOutput&opData={param0:244,param1:"root",param2:"cloupia123",param3:"/bin/date"}

Response

```json
{
  "serviceResult":228,
  "serviceError":null,
  "serviceName":"InfraMgr",
  "opname":"genericvm:userAPIExecuteVIXScriptWithOutput"
}
```

SR Log

```
Service Request ID: 228

Mar 06, 2015 02:08:21 UTC Request submitted
Mar 06, 2015 02:08:22 UTC Executing workflow item number 1
Mar 06, 2015 02:08:22 UTC Completed workflow item number 1, with status Completed
Mar 06, 2015 02:08:23 UTC Executing workflow item number 2
Mar 06, 2015 02:08:23 UTC Trigger context executeWorkflowStep called
Mar 06, 2015 02:08:25 UTC Executing custom action VIX Script (Execute VIX Script)
Mar 06, 2015 02:08:26 UTC Executing custom action VIX Script (Execute VIX Script)
Mar 06, 2015 02:08:32 UTC Task #1 (VIX Script (Execute VIX Script)) completed successfully in 6 seconds
Mar 06, 2015 02:08:32 UTC Input/Output values for Task #1 (VIX Script (Execute VIX Script)):
Mar 06, 2015 02:08:32 UTC [Local Input: Credential Type = Login]
Mar 06, 2015 02:08:32 UTC [Local Input: Login = root]
Mar 06, 2015 02:08:32 UTC [Local Input: Password = "*masked-value*""]
Mar 06, 2015 02:08:32 UTC [Local Input: Script = /bin/date]
Mar 06, 2015 02:08:32 UTC [Local Input: Undo Script = null]
Mar 06, 2015 02:08:32 UTC [Output: EXIT_STATUS_CODE = 0]
Mar 06, 2015 02:08:32 UTC Completed workflow item number 2, with status Completed
```

Input/Output

```
<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task input vmId</td>
<td>244</td>
</tr>
<tr>
<td>Task input credentialType</td>
<td>Login</td>
</tr>
<tr>
<td>Task input login</td>
<td>root</td>
</tr>
<tr>
<td>Task input password</td>
<td>*******</td>
</tr>
<tr>
<td>Task input script</td>
<td>/bin/date</td>
</tr>
<tr>
<td>Task input undoScript</td>
<td>null</td>
</tr>
<tr>
<td>Output</td>
<td>Execute VIX Script.EXECUT_STATUS_CODE = 0</td>
</tr>
<tr>
<td>Output</td>
<td>Execute VIX Script.ERROR_STATUS_MESSAGE = 0</td>
</tr>
</tbody>
</table>
```
Use case 2: Sending an invalid command

Request

```
/app/api/rest?formatType=json&opName=genericvm:userAPIExecuteVIXScriptWithOutput&opData=
{param0:244,param1:"root",param2:"cloupia123",param3:"abcd"}
```

SR Log

```
Service Request ID: 233

Mar 06, 2015 02:30:11 UTC Request submitted
Mar 06, 2015 02:30:12 UTC Executing workflow item number 1
Mar 06, 2015 02:30:12 UTC Completed workflow item number 1, with status Completed
Mar 06, 2015 02:30:12 UTC Executing workflow item number 2
Mar 06, 2015 02:30:12 UTC Executing custom action VIX Script (Execute VIX Script)
Mar 06, 2015 02:30:12 UTC Executing custom action VIX Script (Execute VIX Script)
Mar 06, 2015 02:30:12 UTC Executing custom action VIX Script (Execute VIX Script)
Mar 06, 2015 02:30:12 UTC Executing custom action VIX Script (Execute VIX Script)
Mar 06, 2015 02:30:12 UTC Task #1 (VIX Script (Execute VIX Script)) completed successfully in 9 seconds
Mar 06, 2015 02:30:12 UTC Input/Output values for Task #1 (VIX Script (Execute VIX Script)):
Mar 06, 2015 02:30:12 UTC [Local Input: Select VM = 244]
Mar 06, 2015 02:30:12 UTC [Local Input: Credential Type = Login]
Mar 06, 2015 02:30:12 UTC [Local Input: Login = root]
Mar 06, 2015 02:30:12 UTC [Local Input: Password = **masked-value**]
Mar 06, 2015 02:30:12 UTC [Local Input: Script = abcd]
Mar 06, 2015 02:30:12 UTC [Local Input: Undo Script = null]
Mar 06, 2015 02:30:12 UTC [Output: EXIT_STATUS_CODE = 127]
Mar 06, 2015 02:30:12 UTC [Output: ERROR_STATUS_MESSAGE = sh: abcd: command not found]
Mar 06, 2015 02:30:12 UTC Completed workflow item number 2, with status Completed
```

Input/Output

```
Initiated by admin

VIX Script (Completed)

<table>
<thead>
<tr>
<th>Task input</th>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>vmid</td>
<td></td>
<td>244</td>
</tr>
<tr>
<td>credentialType</td>
<td></td>
<td>Login</td>
</tr>
<tr>
<td>login</td>
<td></td>
<td>root</td>
</tr>
<tr>
<td>password</td>
<td></td>
<td></td>
</tr>
<tr>
<td>script</td>
<td></td>
<td>abcd</td>
</tr>
<tr>
<td>undoScript</td>
<td></td>
<td>null</td>
</tr>
</tbody>
</table>

Output

Execute VIX Script
EXIT_STATUS_CODE
127
ERROR_STATUS_MESSAGE
sh: abcd: command not found
```

Windows VM

Use case 1: Sending a valid command

Request

```
/app/api/rest?formatType=json&opName=genericvm:userAPIExecuteVIXScriptWithOutput&opData=
{param0:247,param1:"administrator",param2:"cloupia123",param3:"C:\\WINDOWS\\system32\\cmd.exe echo hi"}
```
Response

Input/Output

Output file

This API creates an output file (the file extension is .txt) in the file path location /opt/infra/vix_temp_files. For a Windows VM, the API creates a batch file (the file extension is .bat) for verifying the batch command.

```
root@localhost infra:~/vix_temp_files/ root@localhost infra:~/vix_temp_files/ ls -l
viutest-03_06_2015_02_06_26_108.txt viutest-03_06_2015_02_06_26_108.bat
testincludethis vix_temp_files can be used echo hi
```

Multiline commands

Each line is separated by \n.
RestAPIs

Request

```javascript
/app/api/rest?formatType=json&opName=genericvcm:userAPIExecuteVIXScriptWithOutput&opData=
{param0:247,param1:"administrator",param2:"cloupia123",param3:"C:\\WINDOWS\\system32\\cmd.exe
echo hi\n echo this is secondline"
```
Use case 2: Sending an invalid command

Request

/app/api/rest?formatType=json&opName=genericvm:userAPIExecuteVIXScriptWithOutput&opData=
{param0:247,param1:"administrator",param2:"cloupia123",param3:"C:\WINDOWS\system32\cmd.exe whatsapp"}

SR Log

<table>
<thead>
<tr>
<th>Workflow Status</th>
<th>Log</th>
<th>Objects Created and Modified</th>
<th>Input/Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Request ID: 240</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:15 UTC Request submitted</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:21 UTC Executing workflow item number 1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:21 UTC Executing workflow item number 1, with status Completed</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:27 UTC Executing workflow item number 2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:27 UTC Trigger context executeWorkflowStep called</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:28 UTC Executing custom action VIX Script (Execute VIX Script)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:28 UTC Executing custom action VIX Script (Execute VIX Script)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:30 UTC [VIXActionHandler] - account details VMName:windowsvm , Host:172.29.110.75 , OS Type:windows server 2008 r2 (64-bit) , ExitCode:0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC Task #1 (VIX Script (Execute VIX Script)) completed successfully in 3 seconds</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC Input/Output values for Task #1 (VIX Script (Execute VIX Script)):</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC [Local Input: Select VM = 247]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC [Local Input: Credential Type = Login]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC [Local Input: Login = administrator]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC [Local Input: Password = &quot;masked-value&quot;]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC [Local Input: Script = C:\WINDOWS\system32\cmd.exe whatsapp]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC [Local Input: Undo Script = null]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC [Output: EXIT_STATUS_CODE = 9009]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC [Output: ERROR_STATUS_MESSAGE = 'whatsapp' is not recognized as an internal or external command, operable program or batch file.]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:31 UTC Completed workflow item number 2, with status Completed</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:34 UTC Executing workflow item number 3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mar 06, 2015 05:08:34 UTC Executing workflow item number 3, with status Completed</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Input/Output

<table>
<thead>
<tr>
<th>Input/Output</th>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initiated by admin</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VIX Script (Completed)</td>
<td>Task input</td>
<td>vmId</td>
</tr>
<tr>
<td></td>
<td>Task input</td>
<td>credentialType</td>
</tr>
<tr>
<td></td>
<td>Task input</td>
<td>login</td>
</tr>
<tr>
<td></td>
<td>Task input</td>
<td>password</td>
</tr>
<tr>
<td></td>
<td>Task input</td>
<td>script</td>
</tr>
<tr>
<td></td>
<td>Task input</td>
<td>undoScript</td>
</tr>
<tr>
<td></td>
<td>Output</td>
<td>Execute VIX Script:EXIT_STATUS_CODE</td>
</tr>
<tr>
<td></td>
<td>Output</td>
<td>Execute VIX Script:ERROR_STATUS_MESSAGE</td>
</tr>
</tbody>
</table>

WhatsApp is not recognized as an internal or external command, operable program or batch file.
EXECUTE_VIX_SCRIPT (XML based)

Limitations

- Only the status code is rendered. The output message is not displayed.
- To display error messages, use the `userAPIExecuteVIXScriptWithOutput` command.

Linux VM

Use case 1: Sending a valid command

```
```
Use case 2: Sending an invalid command

Generic Sample XML

<table>
<thead>
<tr>
<th>Select VM</th>
<th>Select... SDK-VIX-Dont-Delete_Clone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential Type</td>
<td>Login</td>
</tr>
<tr>
<td>Login</td>
<td>root</td>
</tr>
<tr>
<td>Password</td>
<td>***********</td>
</tr>
<tr>
<td>Script</td>
<td>test</td>
</tr>
</tbody>
</table>

Execute API

UCSD Server: Local UCSD

Execute REST API

Response:

```xml
<xml version="1.0" encoding="UTF-8" standalone="yes">
<log>
<operationStatus>
<response><ExecuteVIXScriptResponse>
<EXIT_STATUS_CODE>1</EXIT_STATUS_CODE>
<ERROR_STATUS_MESSAGE></ERROR Status Message>
<ExitStatusMessage>
<Current Exit Status Code>1</Current Exit Status Code>
</response>
</message>
</log>
</xml>
```

Windows VM

Use case 1: Sending a valid command

Generic Sample XML

<table>
<thead>
<tr>
<th>Select VM</th>
<th>Select... Vioscripttesting_smegnug</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credential Type</td>
<td>Login</td>
</tr>
<tr>
<td>Login</td>
<td>administrator</td>
</tr>
<tr>
<td>Password</td>
<td>***********</td>
</tr>
</tbody>
</table>
| Script          | C:\\Windows\\system32\\cmd.exe mdir C:\\\test\\mapl
```
Use case 2: Sending an invalid command

```
C:\WINDOWS\system32\cmd.exe test
```
Execute API

UCSD Server: Local UCSD

Execute REST API

Response

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<ucxOperationResponse>
  <Log>
    <Messages>
      <Message>
        <TimeStamp>2015-08-04 22:36:22.965</TimeStamp>
        <Severity>INFO</Severity>
        <Text>/VIKACaccount\test\t - account details VNIName : VIXscripts\testing໐</Text>
      </Message>
      <Message>
        <TimeStamp>2015-08-04 22:36:22.965</TimeStamp>
        <Severity>INFO</Severity>
        <Text>Host : 172.20.110.75 , OS Type : microsoft windows server 2008 r2 (64-bit) , ExitCode : 0</Text>
      </Message>
    </Messages>
  </Log>
  <operationStatus>0</operationStatus>
  <response>
    <ExecuteVIKScriptResponse>
      <EXIT_STATUS_CODE>9009</EXIT_STATUS_CODE>
      <ERROR_STATUS_CODE>-error_message-"test\t is not recognized as an internal or external command, \ninsorable program or batch file."-></ERROR_STATUS_MESSAGE>
    </ExecuteVIKScriptResponse>
  </response>
</ucxOperationResponse>