Scripting Made Easy

Overview

Network engineers are always looking for ways to automate the tasks of maintaining network devices. Logging into network devices to make changes or to upgrade software is tedious, time consuming, and prone to errors. Creating interactive scripts takes the monotony out of maintaining network devices. This document will show how easy it is to create and run CCS scripts. For more detailed information, please review the Infoblox CCS Scripting Guide.

Anatomy of a CCS script

A CCS script consists of several required statements:

- Script: This is the name of your script. For example, 'Script: Set up routing'.
- Script-Filter: This statement defines the devices that the script will run on. For example, 'Script-Filter: $Vendor eq "Cisco" and $Model in ["2811", "2821", "871", "2621XM"]'. $Vendor and $Model in this example are well known attributes. In simple English, this statement translates to 'run this script only on cisco 2811, 2821, 871, and 2621XM devices.
- Action: This is the name of the action. After naming your script and defining the devices, what do you want to do on these devices? For example, 'Action: set router statement' is the name of this action
- Action-Commands: This is the command or series of commands that you want to run on the devices. For example, 'Action-Commands: show vlan'. This means the command 'show vlan' will be run.

A CCS script can have the following optional statements:

- Script Section
  - Script-Description
  - Script-Variables
  - Script-Timeout
  - Script-Login
- Action Section
  - Action-Description
  - Action-Filter
  - Action-Timeout
  - Output-Triggers
- Issue Section
  - Issue-ID
  - Issue-Severity
  - Issue-Template
  - Issue-Filter
  - Issue-Details
  - Issue-Variables
  - Issue-Description
- Trigger Section
  - Trigger
  - Trigger-Template
  - Trigger-Commands
  - Trigger-Filter
  - Trigger-Variables
  - Trigger-Description
  - Output-Triggers

Please review the Infoblox CCS Scripting Guide for more information on the above optional statements

Below is a simple script

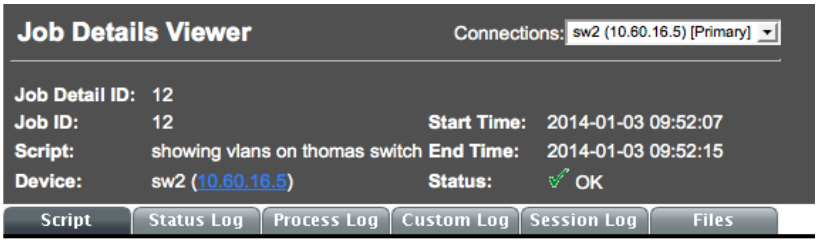
Script: Show vlans

Script-Filter: $Vendor eq “Cisco” and $Model eq “catalyst3560v248ps”

Action: showing vlans

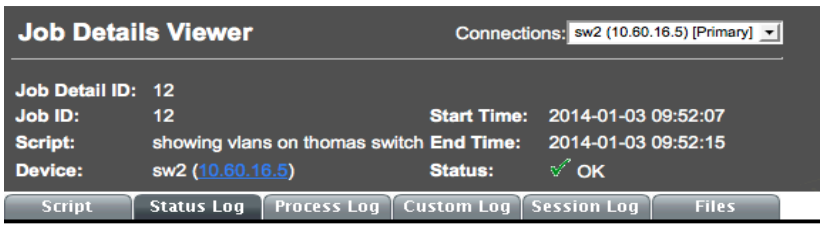
Action-Commands: show vlan

When you run this script, this is what you see:



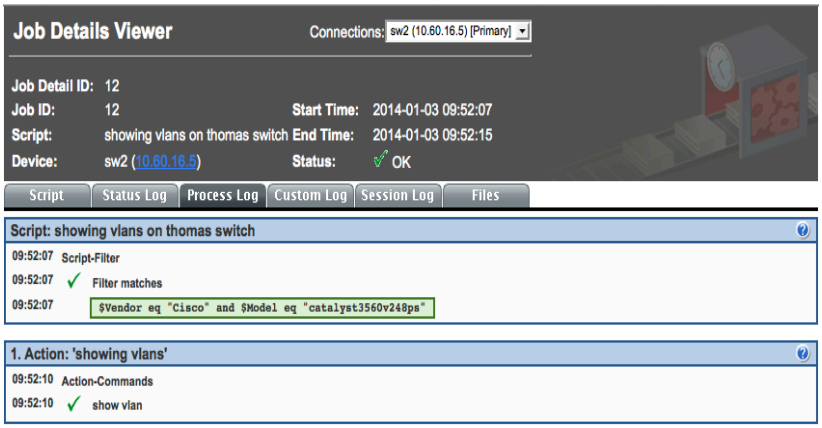

```
#-------------------------------------------------------------------------
# DO NOT EDIT.  This script is maintained by the job specification
# system.  Changing the actual script in the vault or the job
# specification will alter the contents of this copy of the script
# causing a loss of any changes made to the script.  In addition,
# modifying the script file directly will violate any job approval
# mechanism in place.
#-------------------------------------------------------------------------
Script: showing vlans on thomas switch
Script-Description: 'showing vlans on thomas' switch'
Script-Groups:
    MyNetwork: thomas' switch
#-------------------------------------------------------------------------
Script-Filter:  $Vendor eq "Cisco" and $Model eq "catalyst3560v248ps"

Action:  showing vlans

Action-Commands:  show vlan
```

The script tab shows the script

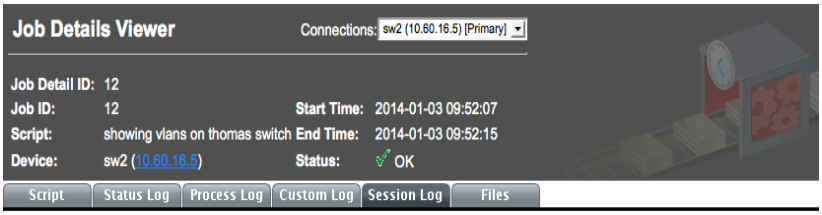



```
+++ Looking up device information ..................................... OK
+++ Looking up device information ..................................... OK
+++ Looking up job specification information .......................... OK
+++ Loading ccs file ................................................. OK

+++ Script: showing vlans on thomas switch
+++ Script-Filter .................................................... MATCH
+++ Looking up authentication information ............................ OK
+++ Opening telnet session with 10.60.16.5 ........................... OK
+++ Got username prompt, sending 'infoblox' ........................... OK
+++ Got password prompt, sending '********' .......................... OK
+++ Prompt detect round [1] ........................................... OK
+++ Prompt type [0] round times [1] [1] ............................... OK
+++ Prompt [sw2>] verify [sw2>] ....................................... OK
+++ Sending 'enable' .................................................. OK
+++ Enable prompt detect round [1] .................................... OK
+++ Got enable password prompt, sending '********' .................... OK
+++ Prompt type [0] round times [1] [1] ............................... OK
+++ Prompt [sw2#] verify [sw2#] ....................................... OK
+++ Sending 'terminal no monitor' ..................................... OK
+++ Sending 'terminal length 24' ...................................... OK
+++ Sending 'terminal no editing' ..................................... OK

+++ 1. Action: showing vlans
+++ 1. [Action-Commands]
+++ 1. Sending 'show vlan' ............................................ OK
+++ Closing session ................................................... OK
*** Successfully ran configuration command script ***
```

The status log tab shows the success status of each part of the process of gathering credentials, using the credentials to log into the device and running the commands. The credentials determined and verified during the discovery process for the device.



The process log tab shows script running. The process log allows you to monitor how well the script ran. If there are any errors that arise during the script run, they would appear here. The check marks show successful execution of each step. In the filter match, the vendor name and model name are case sensitive. You can ensure proper case by looking at the inventory listing for the device.




```
Trying 10.60.16.5...

Connected to 10.60.16.5.

Escape character is '^]'.


User Access Verification

Username: infoblox
Password:
sw2>
sw2>enable
Password:
sw2#
sw2#terminal no monitor
sw2#terminal length 24
sw2#terminal no editing
sw2#show vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa0/2, Fa0/3, Fa0/4, Fa0/5, Fa0/6, Fa0/7, Fa0/8, Fa0/9, Fa0/10, Fa0/11, Fa
2    VLAN0002                         active
5    VLAN0005                         active
9    VLAN0009                         active
10   VLAN0010                         active
11   VLAN0011                         active
12   VLAN0012                         active
13   VLAN0013                         active
19   VLAN0019                         active
20   VLAN0020                         active
21   VLAN0021                         active
26   VLAN0026                         active
28   VLAN0028                         active
30   VLAN0030                         active
39   VLAN0039                         active
40   VLAN0040                         active
41   VLAN0041                         active
```

The session tab shows running of the script from the telnet or SSH session. This session is the same as a manual telnet or SSH session to run the same command.

Let's look at a more complex script that has user input:

Script-Filter:
$Vendor eq "Cisco" and $sysDescr like /IOS/

Script-Variables:

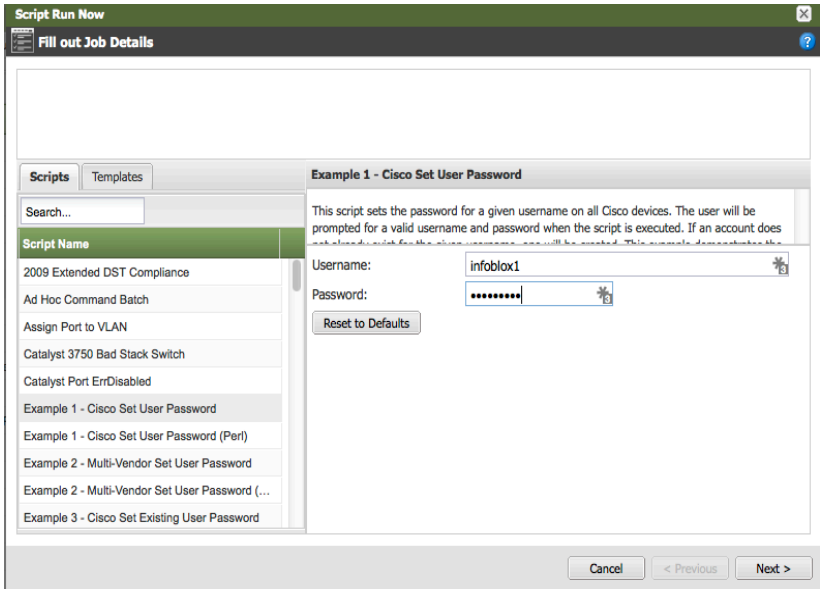
$username	word	"UserName"
$password	password	"New Password"

Action:
Set IOS User Password

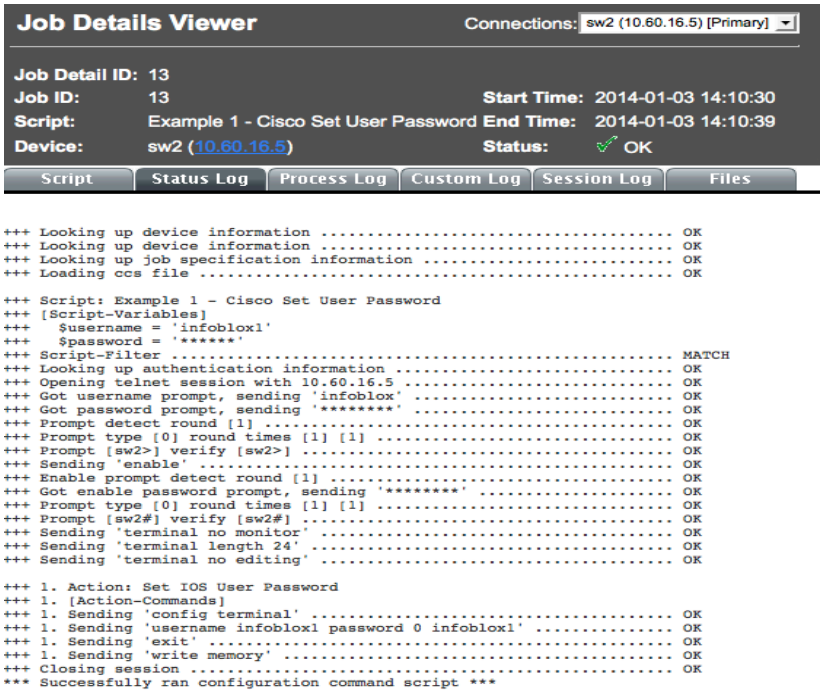
Action-Description:
This action is executed once for every device that matches the Script-Filter defined above. It sets the password for the given user account and then saves the configuration.

Action-Commands:
config terminal
username $username password 0 $password
exit
write memory

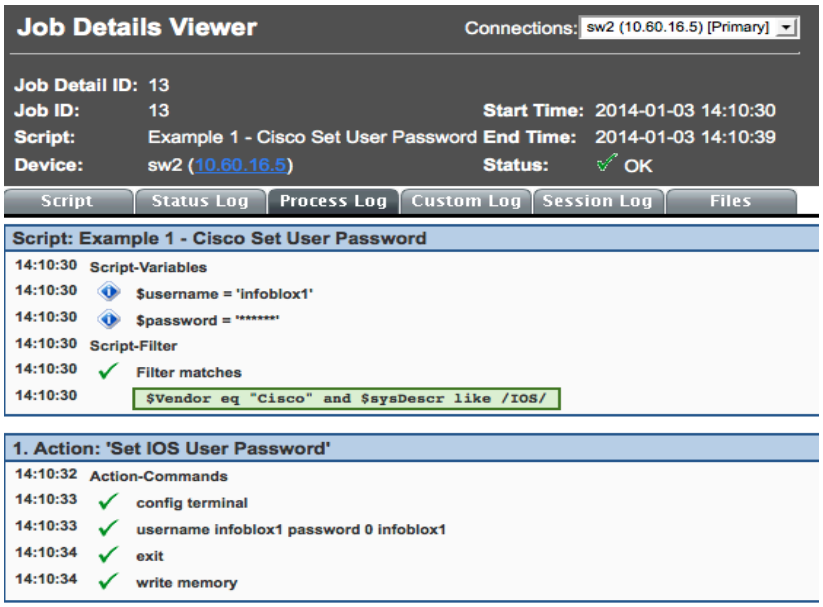
When you run this script, this is what you see:



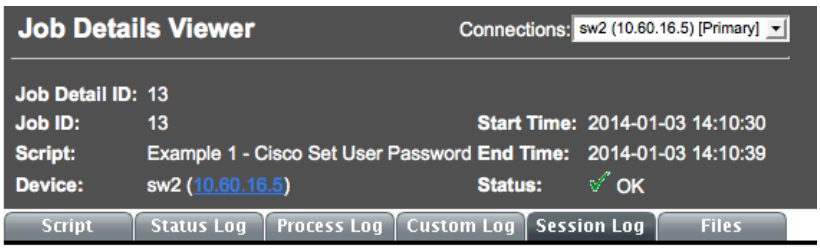
When you run the script, it requests input for the username and password. This input is the result of the definitions of the script variables $username and $password. The username is defined as type "word", which means any string without whitespace can be entered. The password is defined as type "password", which means any string can be entered, however only stars will be displayed. The "UserName" and "Password" fields define the GUI prompts to be displayed to the user when the script is executed.



The status log tab shows the success status of each part of the process of gathering credentials, using the credentials to log into the device and running the commands. The credentials determined and verified during the discovery process for the device.



The process log tab shows script running. The process log allows you to monitor how well the script ran. If there are any errors that arise during the script run, they would appear here. The check marks show successful execution of each step. In the filter match, the vendor name and model name are case sensitive. You can ensure proper case by looking at the inventory listing for the device.




```
Trying 10.60.16.5...

Connected to 10.60.16.5.

Escape character is '^]'.


User Access Verification

Username: infoblox
Password:
sw2>
sw2>enable
Password:
sw2#
sw2#terminal no monitor
sw2#terminal length 24
sw2#terminal no editing
sw2#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
sw2(config)#username infoblox1 password 0 infoblox1
sw2(config)#exit
sw2#write memory
Building configuration...
[OK]
sw2#

*** Job Completed Successfully ***
```

The session tab shows running of the script from the telnet or SSH session. This session is the same as a manual telnet or SSH session to run the same commands.