Raspberry Pi (RPi) Technical Controls

[Bash Cheat Sheet](#)

Topic	Documentation
▷ IP ADDRESS	[Use Case](#) [Command Integration Into Cimitra](#) [Command Contents](#) - Copy/Paste
▷ REBOOT DEVICE	[Use Case](#) [Command Integration Into Cimitra](#) [Command Contents](#) - Copy/Paste

IP ADDRESS

Use Case

A school has RPi devices deployed in every room and even on the outside of the building. The addresses of the RPi devices is not static, they are assigned via DHCP. So discovering an RPi's IP address in order to get an SSH session with the RPi can be difficult since the RPi devices are headless.

Technical Overview

This Cimitra App issues the **ifconfig** Linux operating system command to get the IP address.

Command Contents - Copy/Paste

```
ifconfig
```

Command Integration Into Cimitra

Create a new **Cimitra App** object and fill in the following properties as follows:

CIMITRA APP PROPERTIES

Property	Value
Platform	**Other**
Agent	<The Cimitra Agent deployed to the Raspberry Pi device>
Name	**IP ADDRESS**
Interpreter	**< keep empty >**
Script/Command	`ifconfig`

‹ Back

▷   IP ADDRESS

Platform *

Other   ⇕

Agent *

PI_3B+

Name * (Characters Remaining: 40)

IP ADDRESS

Interpreter

Example: <powershell path>, /bin/bash, /usr/bin/python etc.

Script/Command *

ifconfig

Switches

-c /etc/dbinfo.conf

User Defined Switches / Parameters

+ Add Switch

REBOOT - WITH ACCESS CODE

Use Case

A school has RPi devices deployed in every room and even on the outside of the building. Rebooting an RPi is often the solution to getting everything back into shape. In this scenario, we have added an "**Access Code**" feature so that only those who know the Access Code can actually reboot the Raspberry Pi. It's not a necessary feature, but this scenario explains how to use use the Access Code so only those who know the code can exert the right to actually run the Cimitra App.

Technical Overview

This Cimitra App issues the **reboot** Linux operating system command restart the RPi device.

Command Contents - Copy/Paste

`reboot`

Command Integration Into Cimitra

Create a new **Cimitra App** object and fill in the following properties as follows:

CIMITRA APP PROPERTIES

Property	Value
Platform	**Other**
Agent	\<The Cimitra Agent deployed to the Raspberry Pi device\>
Name	**REBOOT**
Interpreter	\<keep blank\>
Script/Command	`reboot`
User Defined Switches/Parameters	Click the "**+Add Switch**" option. There will be one switch as shown below.

SEND MESSAGE SWITCH

Flag:	\<keep blank\>
Parameter Name:	🔒 **ACCESS CODE**
Validating Regex: The code is **DoIT**	**/^DoIT$/**
Mask	Enable the mask feature so that text isn't shown

Actions | Audit | Users | Agents | Settings | Create ▾

‹ Back

▷ REBOOT DEVICE

Platform *

Other   ⇅

Agent *

PI_3B+

Name * (Characters Remaining: 37)

REBOOT DEVICE

Interpreter

Example: <powershell path>, /bin/bash, /usr/bin/python etc.

Script/Command *

reboot

Switches

-c /etc/dbinfo.conf

User Defined Switches / Parameters

+ Add Switch

🔒 ACCESS CODE 🗑

Flag:	E.G. -c
Parameter Name:	🔒 ACCESS CODE
Validating Regex:	/^DoIT$/
Example:	E.G. $uper$ecret
Mask:	☑ (Like a password)