Contents

1	Intro	oduction	2
2	RES	T requests	3
	2.1	CEE Creation	3
	2.2	Application id retrieval	5
	2.3	Deployment Document	5
	2.4	Start an application	6
	2.5	Stop an application	7
3	Web	User Interface	10
	3.1	Login	10
	3.2	New User Registration	10
	3.3	List CEEs Page	13
	3.4	New CEE Creation	13
	3.5	Edit CEE	15

Chapter 1

Introduction

This document explains how to use VEP from a user point of view. In the next chapter will be explained how to create a CEE, how to set up your application and how finally tostart and stop it.

Before to starts, it is better to explain some VEP elements and rules. CEE stands for Constrained Execution Environment. It is the document that describes the whole environment where the applications run. It contains all the allowed handlers, constraints and rules choosen to deploy the applications. The Deployment Document is a part of an application document and it is sent to the VEP to add VM to the application. The Application Document is the document that describes the complete application, ut contains all the OVF, VMs, reservations and is often send inside the CEE.

A usual operation flow in VEP should follow these steps:

- Creation of a CEE with an Application Document inside ??
- Retriving of the application id
- Send the Deployment Document
- Start the Application
- Stop the Application

The VEP web user interface is still on development so to act all the operations the user should use a rest client tool. There is the RESTClient tools that is compatible with many browser that can be downloaded at http://restclient.net.

Chapter 2

REST requests

2.1 CEE Creation

You can find a description of the fields of a CEE creation request in table 2.1. Here is an example of what such a request could look like.

```
HEADERS:
X-Username:<username>
Content-Type:application/json
BODY:
{
    "name": "new test",
    "state": "active",
    "VMHandlers": [
        {
            "href": "http://vep.fr/vmhandlers/1"
        },
        {
            "href": "http://vep.fr/vmhandlers/3"
        }
    ],
    "applications": [
        {
            "name": "ApplicationSLA",
```

CHAPTER 2. REST REQUESTS

```
"OVFDeployment":false,
        "OVFDescriptors": [
            {
               "OVFFile": "< Escaped String of the OVF>"
            }
        ]
   }
],
"defaultMapping": [
    {
        "type": "VM",
        "virtualSystem": {
            "href": "#ubu1"
        },
        "handler": {
            "href": "http://vep.fr/vmhandlers/3"
        },
        "corecount": 1,
        "ram":256,
        "cpuFreq":500
   }
],
"constraintsMapping":[
  {
      "constraint":
          {"href":"http://vep.fr/constraints/2"},
      "parameter":"FR",
      "virtualSystem":[
          {"href":"#ubu1"}
      ]
  }
]
```

You should retrieve the cee id of the new CEE when it is actually registered.

2.2 Application id retrieval

Once the CEE is created, the user needs to retrieve the also newly created application's id, so he can then update the by default empty application with the virtual machines to deploy. To retrieve this id, it is necessary to do a GET on http://<site>/api/cimi/CEE/{ceeId}/applications. You can find a description of the fields of the response in table 2.2. A sample response could be as in the following example.

HEADERS:

```
X-Username:<username>
Content-Type:application/json
BODY:
{
    "resourceURI": "VEP/AppCollection",
    "id": "http://<site>/cee/{ceeId}/applications",
    "count": 1,
    "applications": [
        {
            "href": "http:/<site>/cee/{ceeId}/application/{appId}",
            "name": "ApplicationSLA",
            reservations: [{reservationId:<reservationId>}]
            //Only if there are any reservations linked to this app
        }
]
```

As described in the formal table, appId is the id which will be needed in the following request to upload the deployment document.

2.3 Deployment Document

To actually create virtual machines from the CEE, the user needs to send a deployment document using a POST request with the following formatting at

CHAPTER 2. REST REQUESTS

this address:

POST http://<site>/api/cimi/cee/<ceeid>/application/<applicationid>. You can find a description of the required fields of the HTTP request in table 2.3. An example follows.

```
HEADERS :
X-Username:<username>
Content-Type:application/json
BODY
{
  "VMs": [
    {
        "name": "AppServer-1",
        "virtualSystem": {"href":"#AppServer"},
        "contextualization": {
                     "key":"value",
                     "key2":"value2"
         }
    },
    {
        "name": "AppServer-2",
        "virtualSystem": { "href": "#AppServer"}
     },
    {
        "name": "DBServer-1",
        "virtualSystem": { "href": "#DBServer"}
    }
  ]
```

2.4 Start an application

To start an application a user needs to send a POST request at the following address : http://<site>/api/cimi/CEE/{ceeId}/application/{appId}/ action/start. Expected result is HTTP/1.1 200 OK only.

2.5 Stop an application

To stop an application a user needs to send a POST request at the following address : http://<site>/api/cimi/CEE/{ceeId}/application/{appId}/ action/stop. Expected result is HTTP/1.1 200 OK only.

name	CEE Identification	name (mandatory	7)
state	'active' or 'check' (mandatory)	
WMH242	Array that lists the	vmhandlers avai	lable in the CEE (mandatory)
	href	Uri of the VMh	undler (mandatory)
StorageHandlers	As VMHandlers (n	ot mandatory)	
NetworkHandlers	As VMHandlers (n	ot mandatory)	
	Array of an applice	ation	
	name	Application nan	ne (mandatory)
	OVFDeployment	always false (ma	undatory)
	OVED accuintance	Array of OVF D	ocument (mandatory)
onoitonilano	O V T.DESCTIPIOIS	OVFFile	Ovf document parsed by an escape function (mandatory)
appincations		Array listing the	e reservations
		mintor Cristom	Reference to the VS in the OVF (mandatory)
٩	reservations	Illiaisystemi	href /# + virtualSystem name' (e.g. #vs1) (mandatory)
2 -		count	Number of VMs to reserve
		enddate	End date of the reservation (mandatory)
	Array mapping ha	ndlers to virtual s	ystems
	type	Only "VM" is s	upported (mandatory)
	corecount	Specify number	of cores for this virtualSystem)
dafan 14 Marina	ram	Specify amount	of memory for this virtualSyste)
Actautuviapping	cpufreq	Specify cpu freq	uency for this virtualSystem)
	vi rtu al Custam	Reference to the	virtual system in the ovf (mandatory)
	ATTIMATO y SUCILI	href	Reference to the VS in the OVF (mandatory)
	handler	href	Uri of the handler (mandatory)
	Array listing the cc	instraints to use in	n the CEE.
	parameter	If there is a cons	straint about country, it must contain the related country code
constraintsManning	constraint	The constraint t	o apply (mandatory)
Gunddanmennennen		href	Uri of the constraint (mandatory)
	xi rtu al System	Array of virtual	System which have to respect the constraint (mandatory)
		href	<pre>'# + virtualSystem name' (e.g. #vs1) (mandatory)</pre>

 Table 2.1: CEE Creation request description

CHAPTER 2. REST REQUESTS

2.5. STOP AN APPLICATION

Table 2.2: Application retrieval response description

resourceURI	VEP/AppCol	lection				
id	http:// <site< th=""><th>e>/cee/<ceeid>,</ceeid></th><th>/applications"</th></site<>	e>/cee/ <ceeid>,</ceeid>	/applications"			
	Array listing	the applications (only one exists by default)				
	href	URL of the application, needed for next step				
applications	name	Name of the application				
	reservations	Array listing the reservations linked to this app, if a				
	reservations	reservationId	an integer identifying a reservation			

Table 2.3: Deployment Document description

	List of VMs of an application						
	name	Name of the VM					
VMe	wirtual System	Virtual System this VM refers to					
V 1V15	viituaiSystem	href /# + virtualSystem name' (e.g. #vs1) (mandatory)					
	contextualization	Dictionary of key:value to transmit to the VM					
	CONTEXTUALIZATION	'key' 'value'(key and value chosen by user)					

Chapter 3

Web User Interface

VEP 2.1 introduces new feautures to make the user experience easier; one is the new Web User Interface that let the user to manage completely his CEEs and Application directly from the browser. In the next line this document describes how to use the web user interface, from "sign in" to "start application"

3.1 Login

To log into VEP(Fig 3.1) it's mandatory to have an account. Every user has to sign in, as described in 3.2 and have a certificate released by the VEP's Administrator. The user has to include his certificate in the browser and then he can log in using username and password.

3.2 New User Registration

As stated in 3.1, to log into VEP it's mandatory to have an account. To register a new user click on "Sign In" on the VEP home page and then (Fig 3.2) insert all the required information.

3.2. NEW USER REGISTRATION

	KEP	
	Please sign in	
	username	
	Password	
	Sign in	
	Sign UP	
contrail	This software is released under BSD license and is free to use. Contrail project is	Contrail consortium
open computing infrastructures services	funded by European Commission under FP7 257438 directive. The source code for Contrail VEP software can be downloaded from OW2 repository.	Contact us Contribute to VEP
Ínnía WENTELISS OU HONDE MURÉEIQUE	VEP REST Web-Interface and the software has been designed by Gaudenzi Filippo, Dudouet Florian and Plyush Harsh with inputs from Yvon Jegou and the collaboration of Semen Marchuk and Ales Cernivec.	Features release timeline Documentation VEP Wiki Download VEP Key developers

Figure 3.1: Login Page

CHAPTER 3. WEB USER INTERFACE

	KEP	
Sign Up		
Username		
Password		
Retype Password		
Email		
	Submit	
	This software is released under BSD license and is free to use. Contrall project is funded by European Commission under FP7 257438 directive. The source code for Contrall VEP software can be downloaded from OW2 repository. VEP REST Web-Interface and the software has been designed by Gaudenzi Filippo, Dudouet Florian and Plyush Harsh with inputs from Yvon Jegou and the collaboration of Semen Marchuk and Ales Cemivec.	Contrall consortium Contact us Contribute to VEP Features release timeline Documentation VEP Wiki Download VEP Key developers

Figure 3.2: Sign In Page

List CEEs Create New CEE Logout	HE!	
	← Edit CEE Create new CEE	
	This software is released under BSD license and is free to use. Contrail project is funded by European Commission under FP7 257435 directive. The source code for Contrail VEP software can be downloaded from OW2 repository. VEP REST Web-interface and the software has been designed by Gaudenzi Filipop. Dudouet Florian and Plyush Harsh with inputs from Yvon Jegou and the collaboration of Semen Marchuk and Ales Cernivec.	Contrail consortium Contact us Contribute to VEP Features release timeline Documentation VEP Wiki Download VEP Key developers

Figure 3.3: List CEEs Page

CEE	name	CEE-1	
CEE	state	ACTIVE	×
M I	handlers SmallHandler		info
'M I	handlers SmallHandler BigHandler		info

Figure 3.4: Create new CEE Page

3.3 List CEEs Page

As you are logged, you will see this page (Fig 3.3). If you have already some CEEs you will see them clicking on "Edit CEE". Choosing one then you can add VMs to it (3.5). As well clicking on "Create New CEE", you can create a new CEE.

3.4 New CEE Creation

This page (Fig 3.4) is made to create a new CEE. You can easy follow the instructions in the form and fill all the required box. Please remember that before to parse the ovf you have tho choose at least one VMHandler. After you have parsed the OVF, you can specify the amount of resource for each Virtual System. Remeber that VEP follows these rules:

- In case you leave the resource fields empty, VEP will use the low-bounds of the choosen VMHandler. Be carefull that in the case that the values specified in the OVF respect the VMhandler's ranges, VEP will use these values.
- If there are values in the resource fields thay will be alway preferred over OVF and VMHandler values but they have to be higher than the OVF values

Reservations and Constrains have to be filled also after you have parsed the OVF, but they are not mandatory to succeed in creating a CEE.

CHAPTER 3. WEB USER INTERFACE

Application name:		Applicatio-1			
حyster حي حي حاکيما حالهه مي حي حي حي حي حي حي حي حي حي حي حي حي حي	n- ssd: ElementName- Virtual Ha ssd: InstanceID-0ssd: VirtualSystemType>vmx- m- asd: Description- <u>Number of vi</u> ssd: ElementName>1 virtual Q ssd: InstanceID>1ssd: ResourceType>3ssd: VirtualQuantitv>1 <th>rdware Family-/vssd nceID> 14-/vssd:VirtualSyste 1ual CPUs-/rasd:De PU-/rasd:ElementNs rceID> esourceType> irtualQuantity></th> <th>:ElementName> mType> scription> mme></th> <th></th> <th></th>	rdware Family-/vssd nceID> 14-/vssd:VirtualSyste 1ual CPUs-/rasd:De PU-/rasd:ElementNs rceID> esourceType> irtualQuantity>	:ElementName> mType> scription> mme>		
		Parse OVF			
/M mapping Virtual System	VM Handler	Parse OVF	J freq.	Cores	RAM
/M mapping Virtual System Reservations	VM Handler	Parse OVF	J freq.	Cores	RAM
/M mapping Virtual System Reservations Virtual System	VM Handler Qty	Parse OVF	J freq.	Cores	RAM
/M mapping Virtual System Reservations Virtual System Constraints	VM Handler Qty	Parse OVF	J freq.	Cores	RAM Del

Figure 3.5: Create new CEE Page - OVF Parsing

Application n	ame:	Applicatio-1	Applicatio-1						
	System> <sset elementnam<br=""><sset instanced=""> <sset th="" virtualsystem<=""><th>le>-Virtual Hardware Fan >/vssd/linstanceID> nType>vmx-04-vssd.Vi Number of virtual CPUs- reard instanceID> pe>3-rinsd:ResourceTy itv=1-virasd:ResourceTy itv=1-virasd:VirtualQuant</th><th>ily tualSystemType> /rasd:Description> lementName> 29> tiv></th><th></th><th></th></sset></sset></sset>	le>-Virtual Hardware Fan >/vssd/linstanceID> nType>vmx-04-vssd.Vi Number of virtual CPUs- reard instanceID> pe>3-rinsd:ResourceTy itv=1-virasd:ResourceTy itv=1-virasd:VirtualQuant	ily tualSystemType> /rasd:Description> lementName> 29> tiv>						
		Pars	e OVF						
VM mapp	ing								
System	VM Handler	CPU freq.	Cores	RAM					
ubu1	VM Handler CPU freq. Cores RAM ubu1 SmallHandler Image: Cores Image: Cores								

Figure 3.6: Create new CEE Page - Binding VM to VMhandler

intual system Count Free									
/irtual macl	nines								
Name	VS handler	VM handler	State	IP	Context	Delete			
machine1	#ubu1	VerySmallHandler	INI	undefined					
machine2	#ubu1	VerySmallHandler	INI	undefined					
machine2	#ubu1	VerySmallHandler	INI	undefined					

Figure 3.7: Edit CEE Page

Name	VS handler	VM handler	State	IP	Context	Delete
machine1	#ubu1	VerySmallHandler	INI	undefined		
machine2	#ubu1	VerySmallHandler	INI	undefined		
	#ubu1			"key1":"val1","key2":"val2",	delete	

Figure 3.8: Edit CEE Page - adding VMs

3.5 Edit CEE

From page (Fig 3.3), choosing a CEE is possible to make operations on the application related to it. The page (3.7 shows a summury of the CEE and its application. From this page you can start the application, stop the application, and add/delete VMs. As you add or delete one or more VMs remember to click on the button "SAVE" to send the command to VEP.