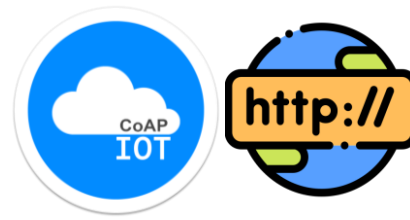


Laboratorio 5:

HTTP y CoAP Servers

Máquina Virtual



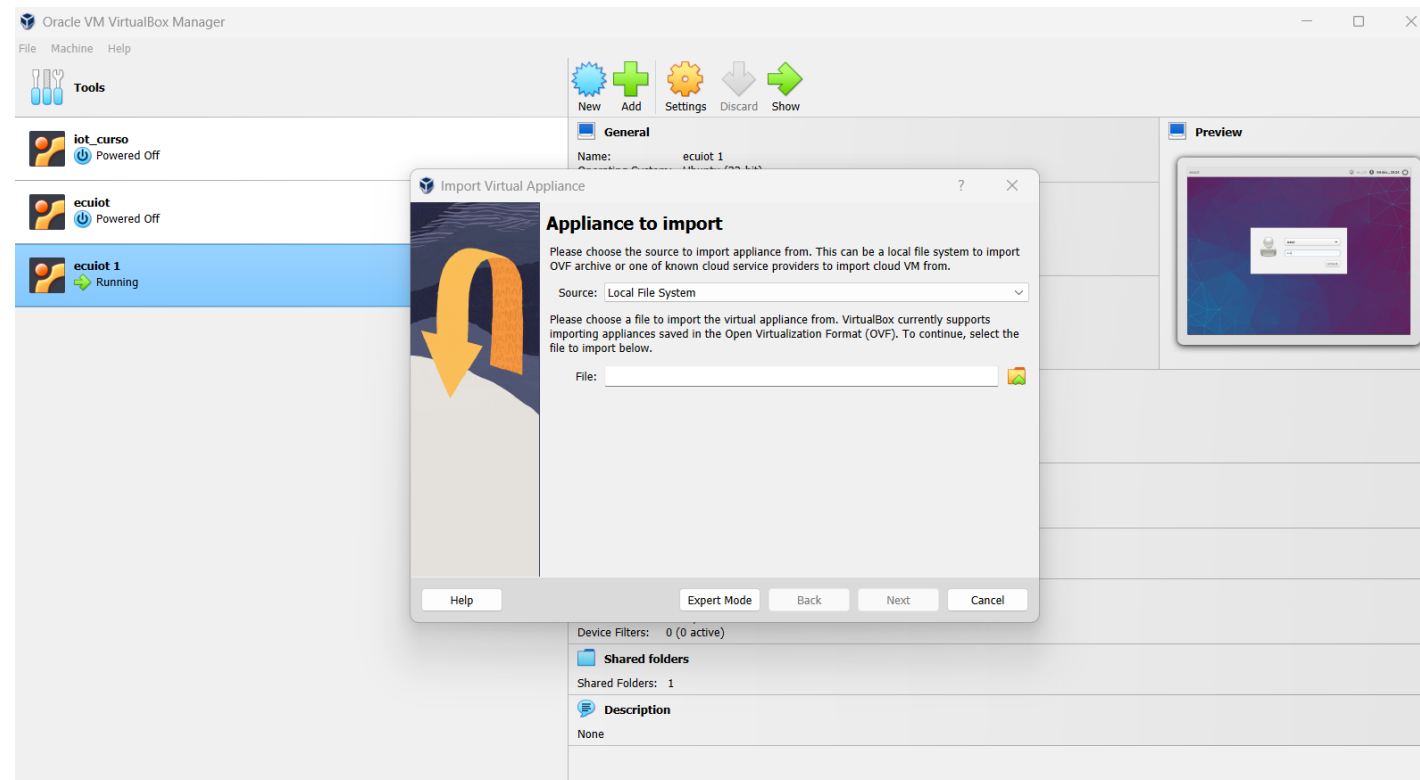
File -> Import Apliance...

File -> dirección de donde se descargó la imagen virtual.

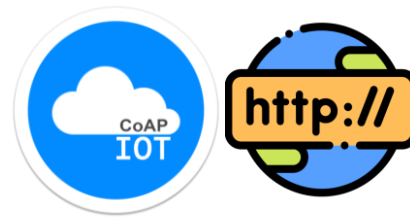
Next ->

Finish ->

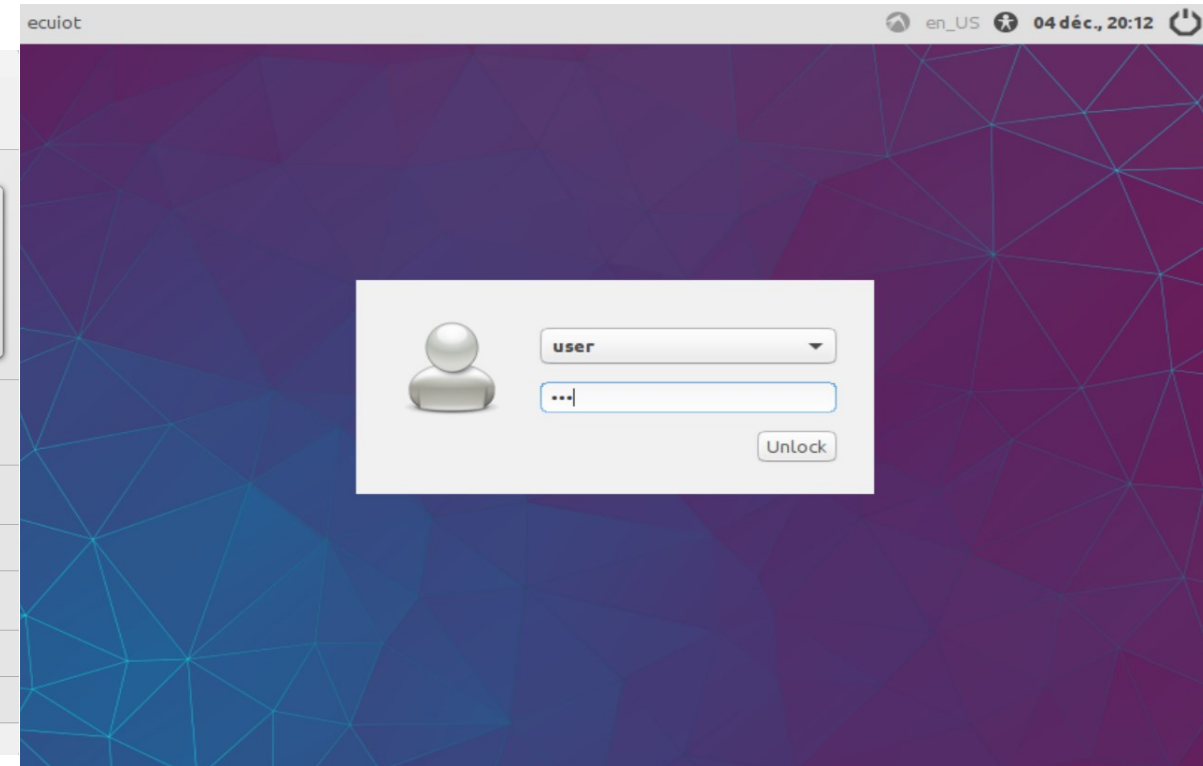
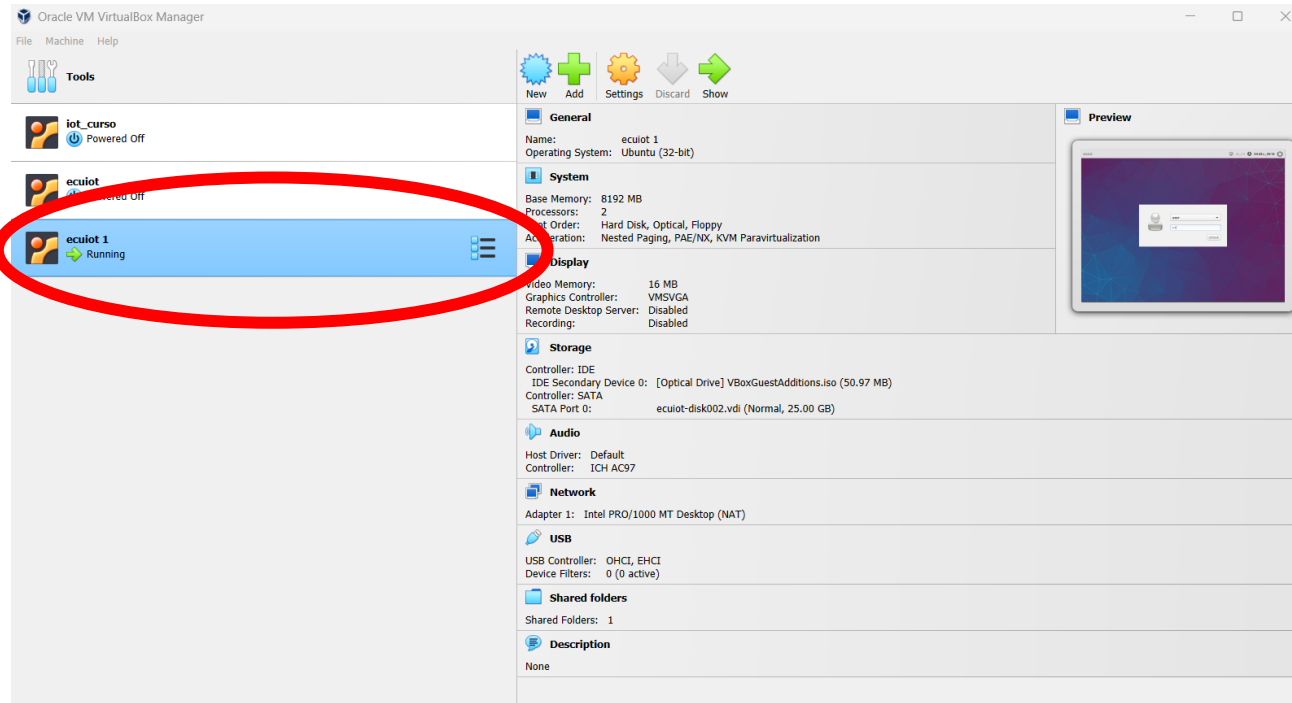
Esperar hasta que la imagen se instale



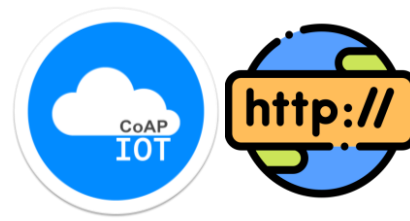
Máquina Virtual



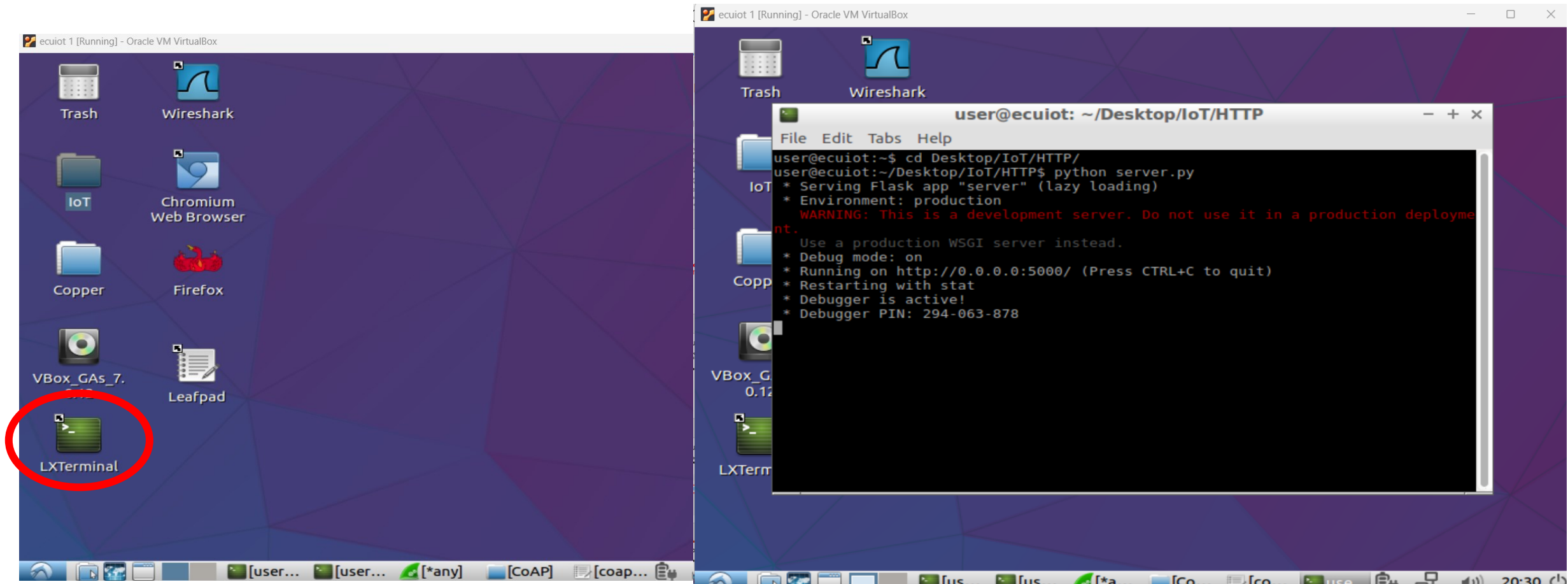
Iniciar Máquina Virtual llamada "ecuiot" desde Oracle VM.
Clave de acceso: 123



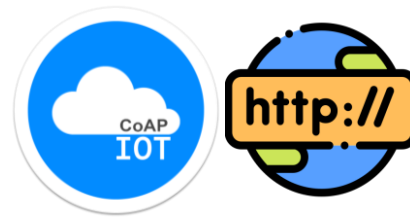
HTTP Servicio Ejemplo



En la terminal acceder carpeta con script de HTTP servicio.
Iniciar el servicio: `python service.py`



HTTP Servicio Ejemplo



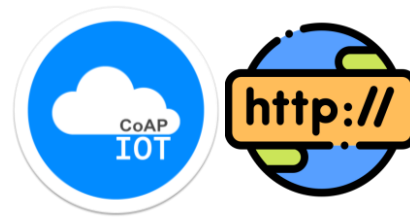
En una terminal nueva se puede interactuar con el servicio creado:
`curl -X get http://0.0.0.0:5000/living_room/temperature`

A screenshot of a terminal window titled 'user@ecuiot: ~' inside an Oracle VM VirtualBox. The terminal shows two curl commands being executed. The first command is `curl -X GET http://0.0.0.0:5000/living_room/temperature`, which results in an error: `curl: (6) Could not resolve host: X-`. The second command is `curl -X GET http://0.0.0.0:5000/living_room/temperature`, which returns the value `24.58`. Below the terminal window, a network log shows a successful GET request: `[04/Dec/2023 20:33:32] "GET /living_room/temperature HTTP/1.1" 200`. The background of the terminal window shows a desktop environment with icons for Trash, folders, and a terminal window titled 'LXTerm'.

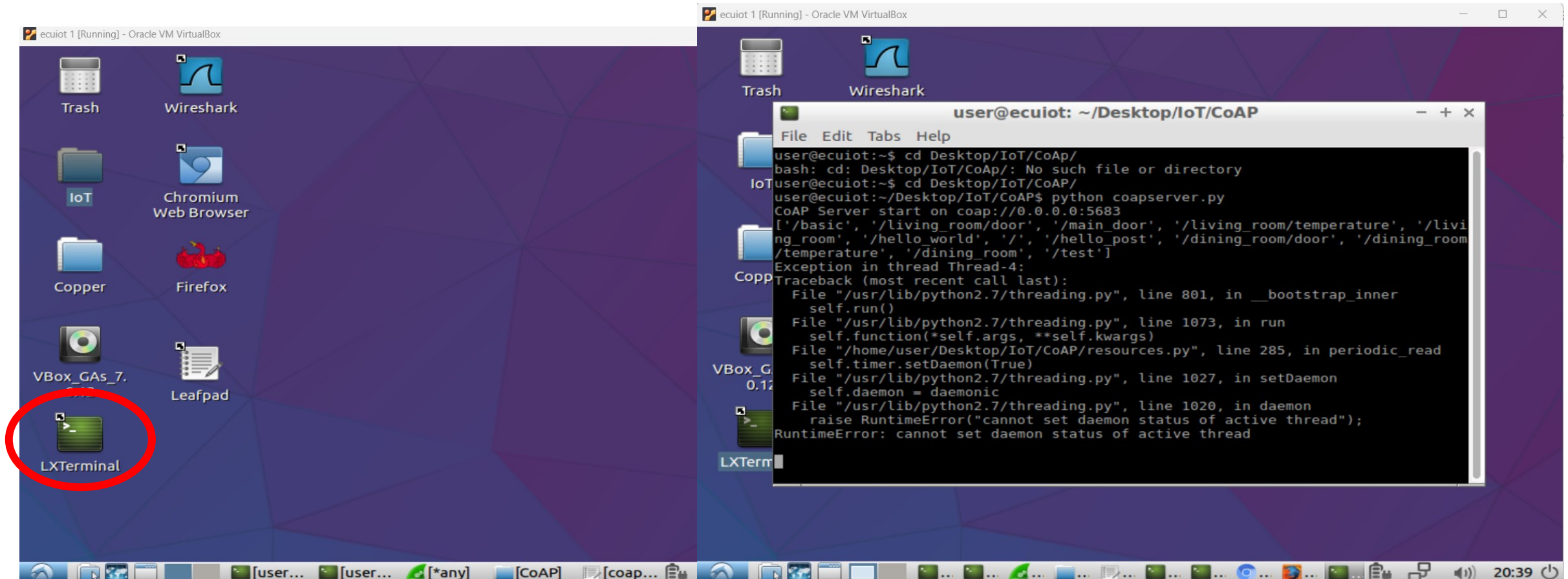
Métodos:

- GET: Lee Recursos
- POST: Crea Nuevos Recursos
- PUT: Actualiza Recursos
- DELETE: Elimina Recursos.

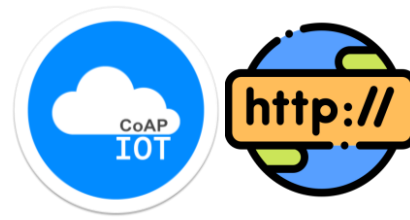
CoAP Servicio Ejemplo



En la terminal acceder carpeta con script de CoAP servicio.
Iniciar el servicio: `python coapserver.py`



CoAP Servicio Ejemplo



En una terminal nueva se puede interactuar con el servicio creado:
`coap get coap://0.0.0.0:5683/living_room/temperature`

A screenshot of a terminal window titled 'user@ecuiot: ~' running on an Oracle VM VirtualBox. The terminal shows a sequence of commands and their outputs. The first command is 'coap://0.0.0.0:5683/living_room/temperature', which results in a 'bash: coap://0.0.0.0:5683/living_room/temperature: No such file or directory' error. The second command is 'coap get coap://0.0.0.0:5683/living_room/temperature', which returns '(2.05) 18.4'. The terminal also shows a list of files in the background, including 'self.', 'File "/', 'self.', 'File "/', 'self.', 'File "/', 'self.', 'File "/', 'self.', 'File "/', 'raise', 'CoppRuntimeEr', '2023-12-6', 'tagram -', 'ing_room,', '2023-12-6', 'VBox_G', '0.12', 'nd_respor', '.4...4 bytes', '2023-12-04 20:41:06,401 - Thread-34 - coapthon.server.coap - DEBUG - send_datag', 'ram - From None, To ('127.0.0.1', 55317), ACK-51056, CONTENT-Údö [] 18.4...4 b', and 'LXTermytes'. The terminal window is overlaid on a desktop environment with a purple background and various icons.

CoAP Servicio Ejemplo con Cupper



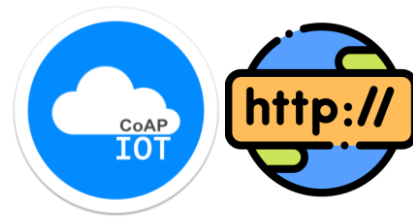
Métodos

Recursos

The screenshot shows the Cupper CoAP client interface in a Mozilla Firefox browser window. The address bar displays `coap://0.0.0.0:5683/living_room`. The interface includes a toolbar with various HTTP methods: Ping, Discover, GET, FETCH, POST, PATCH, PUT, IPATCH, DELETE, and Observe. The 'Observe' method is highlighted with a red circle. On the left, a 'Discovering' tree shows the resource structure, with 'living_room' selected and also circled in red. The main area displays the 'Petición/Respuesta' (Request/Response) details. The bottom section shows a table of CoAP messages:

T...	CoAP Message	...	Token	Payload
21:10:54	CON-GET	15693 (0)	empty	
21:10:54	ACK-2.05 Content	15693	empty	</basic>;rt="rt1";if="if1",</living_room/door>;</main_door>;</living_room/temperature>;obs,</living_room...

CoAP Servicio Ejemplo con Cupper



DISCOVER: Muestra los recursos disponibles.

GET: Lee el valor de los recursos.

POST/PULL: Cambia el valor de un recurso.

POST: Crea un nuevo recurso.

DELETE: elimina recursos.

OBSERVE: Obtiene valores de recursos cada vez que cambia.

CoAP con Wireshark inspección de paquetes



- **coap.code == X**

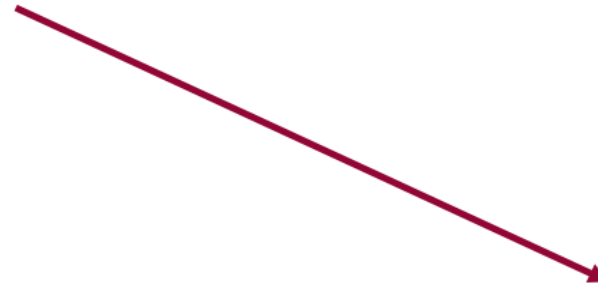
X puede ser el método requerido



Code	Name	Reference
1	GET	[RFCXXXX]
2	POST	[RFCXXXX]
3	PUT	[RFCXXXX]
4	DELETE	[RFCXXXX]

CoAP Method Codes

X puede ser el código de respuesta



coap.type == Y

Y puede ser el tipo de mensaje

0 = CON

1 = NON

2 = ACK

3 = RST

Code	Description	Reference
65	2.01 Created	[RFCXXXX]
66	2.02 Deleted	[RFCXXXX]
67	2.03 Valid	[RFCXXXX]
68	2.04 Changed	[RFCXXXX]
69	2.05 Content	[RFCXXXX]
128	4.00 Bad Request	[RFCXXXX]
129	4.01 Unauthorized	[RFCXXXX]
130	4.02 Bad Option	[RFCXXXX]
131	4.03 Forbidden	[RFCXXXX]
132	4.04 Not Found	[RFCXXXX]
133	4.05 Method Not Allowed	[RFCXXXX]
134	4.06 Not Acceptable	[RFCXXXX]
140	4.12 Precondition Failed	[RFCXXXX]
141	4.13 Request Entity Too Large	[RFCXXXX]
143	4.15 Unsupported Media Type	[RFCXXXX]
160	5.00 Internal Server Error	[RFCXXXX]
161	5.01 Not Implemented	[RFCXXXX]
162	5.02 Bad Gateway	[RFCXXXX]
163	5.03 Service Unavailable	[RFCXXXX]
164	5.04 Gateway Timeout	[RFCXXXX]
165	5.05 Proxying Not Supported	[RFCXXXX]

CoAP Response Codes