

UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE CIENCIAS



ÁREA DE CIENCIA DE LA COMPUTACIÓN
10° LABORATORIO - CC312

- **TÍTULO:** Explorar los modelos YANG

- **ALUMNO:**

JHONATAN FLORINS POMA MARTINEZ

20182729F

- **PROFESORES:** YURI JAVIER.,CCOICCA PACASI

2023

Objetivos

Parte 1: Iniciar la máquina virtual de DEVASC.

Parte 2: Explorar un modelo de YANG en GitHub.

Parte 3: Explorar un modelo YANG usando pyang.

Introducción:

YANG (Yet Another Next Generation) es un lenguaje de modelado utilizado en el ámbito de la gestión de redes y dispositivos de red. Los modelos de datos YANG se utilizan para intercambiar información entre los dispositivos de red y los sistemas de gestión, lo que permite una configuración automatizada, el monitoreo y la operación de la red.

YANG se basa en una sintaxis similar a la de XML y utiliza una estructura basada en árboles para representar la jerarquía de los datos. Además, YANG permite definir tipos de datos personalizados, restricciones y relaciones entre los elementos de datos. Esto facilita la interoperabilidad y la consistencia en la gestión de redes al establecer una especificación clara y común de los datos que se pueden intercambiar entre diferentes dispositivos y sistemas de gestión.

Pyang es una herramienta de línea de comandos utilizada para validar y analizar modelos YANG. Se utiliza para verificar la sintaxis y la semántica de los modelos YANG, asegurándose de que cumplan con las reglas y restricciones definidas en el estándar YANG. Además, Pyang puede generar información detallada sobre los modelos, incluyendo árboles de datos, diagramas de dependencia y documentación legible para los humanos.

Parte 1: Iniciar la máquina virtual de DEVASC.




Iniciado.

Parte 2: Explorar un modelo de YANG en GitHub.

En esta parte, instalará el módulo pyang en su máquina virtual DEVASC y explorará como transforma los archivos YANG. Pyang simplifica el trabajo con archivos YANG. El módulo viene con una línea ejecutable de comandos pyang que transforma los archivos YANG en un formato más legible para el ser humano.

Paso 1: Explorar los modelos de Cisco IOS XE YANG en el repositorio de GitHub.

En la rama maestra, vaya a los modelos de YANG para Cisco IOS XE (versión 16.9.3), haciendo clic en los siguientes directorios: **vendor > cisco > xe > 1693 > ietf-interfaces.yang**

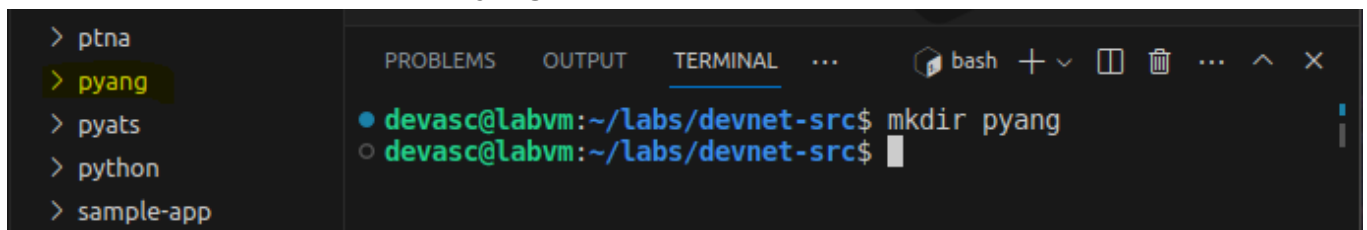
| | | |
|--|---|-------------|
|  ietf-interfaces-ext.yang | Added IOS XE 16.9.3 yang models and fixed IOS XE 16.... | 4 years ago |
|  ietf-interfaces.yang | Added IOS XE 16.9.3 yang models and fixed IOS XE 16.... | 4 years ago |
|  ietf-ip.yang | Added IOS XE 16.9.3 yang models and fixed IOS XE 16.... | 4 years ago |

Haga clic en `ietf-interfaces.yang` y desplácese por todos los nodos de contenedores. Por ejemplo, alrededor de la línea 221 verá la hoja habilitada. (cuando usemos `pyang` en la parte 3, veremos que se ve mas ordenado en formato árbol)

```
221     leaf enabled {
222         type boolean;
223         default "true";
224         description
225             "This leaf contains the configured, desired state of the
226             interface.
227
228             Systems that implement the IF-MIB use the value of this
229             leaf in the 'running' datastore to set
230             IF-MIB.ifAdminStatus to 'up' or 'down' after an ifEntry
231             has been initialized, as described in RFC 2863.
232
233
234             Changes in this leaf in the 'running' datastore are
235             reflected in ifAdminStatus, but if ifAdminStatus is
236             changed over SNMP, this leaf is not affected.";
237         reference
238             "RFC 2863: The Interfaces Group MIB - ifAdminStatus";
239     }
240 }
```

Paso 2: Copiar el modelo `ietf-interfaces.yang` en una carpeta de su Máquina Virtual.

Creemos un subdirectorio llamado **pyang** en el directorio **devnet-src**.

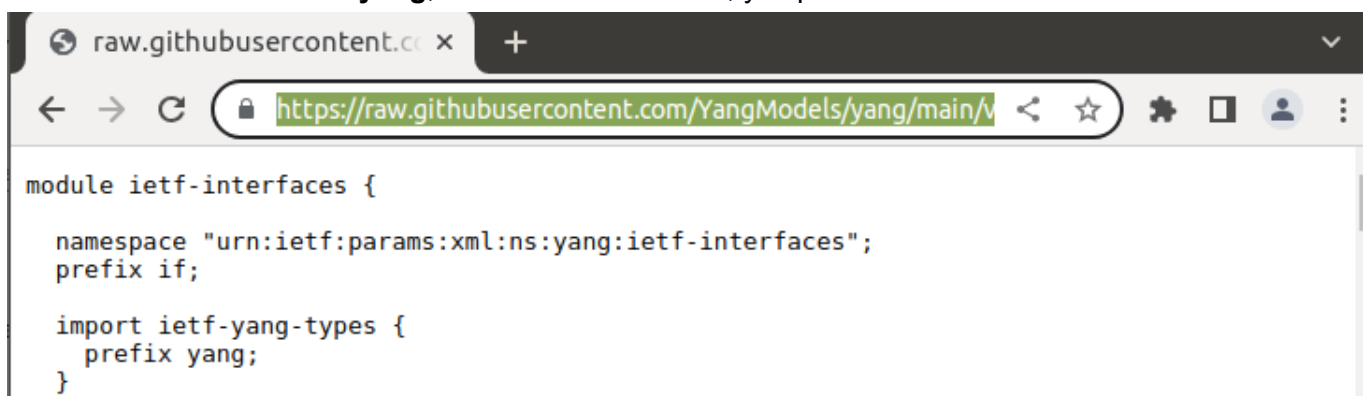


The screenshot shows a terminal window with a file explorer on the left and a terminal on the right. The file explorer shows a directory structure with 'pyang' highlighted. The terminal shows the command 'mkdir pyang' being executed successfully.

```
> ptna
> pyang
> pyats
> python
> sample-app
```

```
PROBLEMS OUTPUT TERMINAL ...
devasc@labvm:~/labs/devnet-src$ mkdir pyang
devasc@labvm:~/labs/devnet-src$
```

Del archivo **ietf-interfaces.yang**, hacemos click en **raw**, y copiamos el URL



The screenshot shows a web browser window with the URL 'https://raw.githubusercontent.com/YangModels/yang/main/ietf-interfaces.yang'. The page displays the raw content of the YANG model file, which is a text-based configuration.

```
module ietf-interfaces {
    namespace "urn:ietf:params:xml:ns:yang:ietf-interfaces";
    prefix if;

    import ietf-yang-types {
        prefix yang;
    }
}
```

En la terminal usamos **wget** + URL para guardar el archivo **ietf-interfaces.yang** sin procesar.

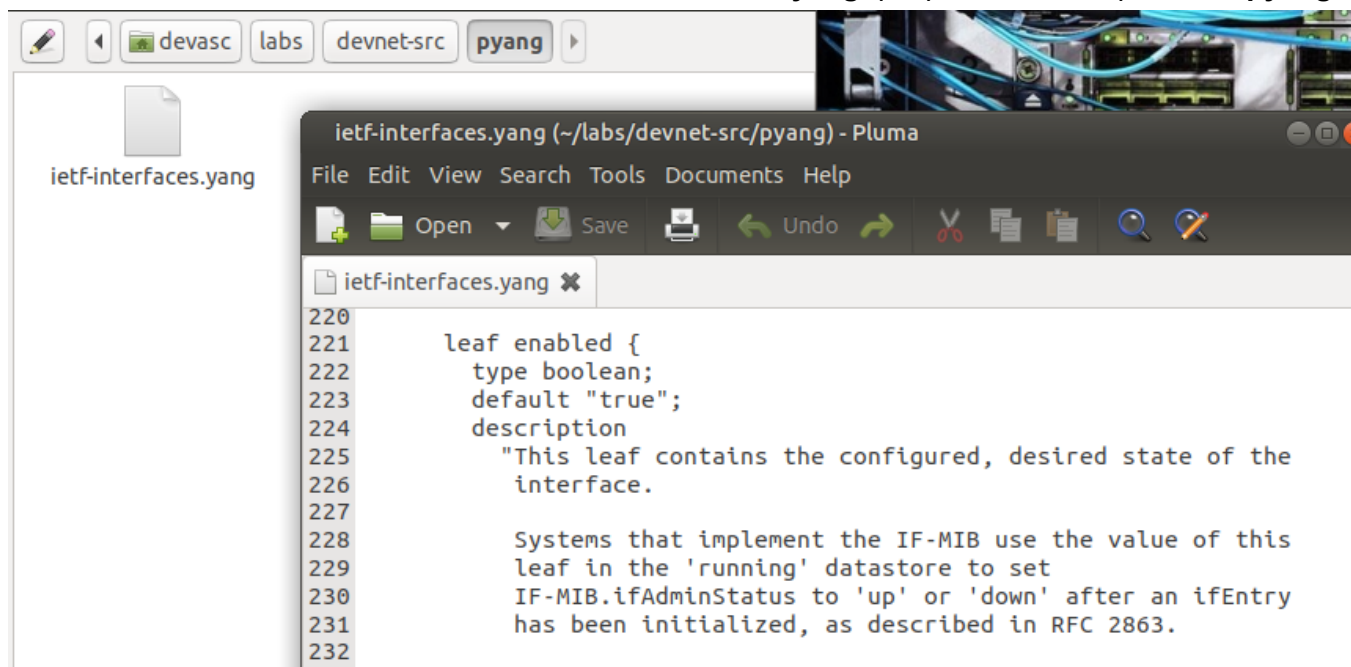
```
devasc@labvm:~/labs/devnet-src$ mkdir pyang
devasc@labvm:~/labs/devnet-src$ cd pyang
devasc@labvm:~/labs/devnet-src/pyang$ wget https://raw.githubusercontent.com/YangModels/yang/main/vendor/cisco/xr/1693/ietf-interfaces.yang
--2023-06-15 23:54:02-- https://raw.githubusercontent.com/YangModels/yang/main/vendor/cisco/xr/1693/ietf-interfaces.yang
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.108.133, 185.199.110.133, 185.199.111.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 24248 (24K) [text/plain]
Saving to: 'ietf-interfaces.yang'

ietf-interfaces.yan 100%[=====] 23.68K --.-KB/s in 0.01s

2023-06-15 23:54:02 (2.30 MB/s) - 'ietf-interfaces.yang' saved [24248/24248]

devasc@labvm:~/labs/devnet-src/pyang$
```

Ahora tenemos una versión local del modelo **ietf-interfaces.yang** que podemos manipular con **pyang**.



Parte 3: Explorar un modelo YANG usando pyang.

En esta parte, usted instalará el módulo **pyang** en su máquina virtual DEVASC y explorará cómo transforma el modelo YANG que copió de GitHub. Pyang simplifica el trabajo con archivos YANG. El módulo viene con un ejecutable de línea de comandos **pyang** que transforma los archivos YANG en un formato más legible por humanos.

Paso 1: Verificar que **pyang** esté instalado y actualizado.

```
devasc@labvm:~/labs/devnet-src$ pyang -v
pyang 2.5.3
devasc@labvm:~/labs/devnet-src$ pip3 install pyang --upgrade
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: pyang in /home/devasc/.local/lib/python3.8/site-packages (2.5.3)
Requirement already satisfied: lxml in /home/devasc/.local/lib/python3.8/site-packages (from pyang) (4.5.1)
devasc@labvm:~/labs/devnet-src$
```

Paso 2: Transformar el modelo ietf-interfaces.yang.

Ingresa **pyang -h | more** para explorar más las opciones para transformar el modelo YANG. Busque la opción **-f** como se muestra a continuación. Usaremos la opción de formato de árbol.

```
devasc@labvm:~/labs/devnet-src/pyang$ pyang -h | more
Usage: pyang [options] [<filename>...]

Validates the YANG module in <filename> (or stdin), and all its dependencies.

Options:
-h, --help                Show this help message and exit
-v, --version             Show version number and exit
-V, --verbose
-e, --list-errors         Print a listing of all error and warning codes and
                           exit.
-f FORMAT, --format=FORMAT
                           Convert to FORMAT. Supported formats are: yang, yin,
                           dsdl, identifiers, tree, name, uml, depend, omni,
                           sample-xml-skeleton, capability, jsonxsl, jstree,
                           jtox, flatten
-o OUTFILE, --output=OUTFILE
                           Write the output to OUTFILE instead of stdout.
```

Transformamos el modelo **ietf-interfaces.yang** en un modelo de **formato de árbol** con el siguiente comando. Observe que la **hoja enabled** es mucho más fácil de encontrar y leer en este formato.

```
devasc@labvm:~/labs/devnet-src/pyang$ pyang -f tree ietf-interfaces.yang
ietf-interfaces.yang:6: error: module "ietf-yang-types" not found in search path
module: ietf-interfaces
+--rw interfaces
|   +--rw interface* [name]
|   |   +--rw name                string
|   |   +--rw description?        string
|   |   +--rw type                identityref
|   |   +--rw enabled?            boolean
|   |   +--rw link-up-down-trap-enable? enumeration {if-mib}?
+--ro interfaces-state
+--ro interface* [name]
+--ro name                string
+--ro type                identityref
+--ro admin-status        enumeration {if-mib}?
+--ro oper-status         enumeration
+--ro last-change?        yang:date-and-time
+--ro if-index            int32 {if-mib}?
+--ro phys-address?       yang:phys-address
+--ro higher-layer-if*    interface-state-ref
+--ro lower-layer-if*     interface-state-ref
+--ro speed?              yang:gauge64
+--ro statistics
+--ro discontinuity-time   yang:date-and-time
+--ro in-octets?           yang:counter64
+--ro in-unicast-pkts?     yang:counter64
+--ro in-broadcast-pkts?   yang:counter64
+--ro in-multicast-pkts?   yang:counter64
+--ro in-discards?        yang:counter32
+--ro in-errors?          yang:counter32
+--ro in-unknown-protos?   yang:counter32
+--ro out-octets?          yang:counter64
+--ro out-unicast-pkts?    yang:counter64
+--ro out-broadcast-pkts?  yang:counter64
+--ro out-multicast-pkts?  yang:counter64
+--ro out-discards?       yang:counter32
+--ro out-errors?         yang:counter32
```

Conclusiones:

Usamos pyang para explorar el archivo "ietf-interfaces.yang", y se transformó el modelo en un formato de árbol utilizando el comando "pyang -f árbol ietf-interfaces.yang". Esto facilitó la lectura y comprensión del modelo, destacando la hoja "enabled" de manera más clara.

- La herramienta pyang facilita el trabajo con archivos YANG al proporcionar una forma más legible para los humanos de visualizar y transformar estos archivos, permitiendo a los desarrolladores y administradores de redes trabajar con estos archivos de manera más eficiente y comprender su estructura.