

```

from fireREST import FMC
import json

###Global Variable

def exportaCSV(fmc, ac_policy):

    reglas = fmc.policy.accesspolicy.accessrule.get(ac_policy)

    policyFile = open(ac_policy + ".csv", "w")
    policyFile.write("#, name,sourceZones, destZones,
sourceNetworks, destNetworks, sourcePorts,destPorts, URL, APP, VLAN,
Action, Enable, ipsPolicy, variableSet, filePolicy, logBegin,
logEnd, sendEventsToFMC, syslogConfig, TimeRangeObject, Comments\n")

    contador = 0
    # imprimir en json todo --> print(json.dumps(reglas,indent=2))

    for ele in reglas:
        #print("#####")
        contador = contador + 1

        try:
            ips = ele['ipsPolicy']['name']
        except KeyError:
            ips = "NULL"

        try:
            file = ele['filePolicy']['name']
        except KeyError:
            file = "NULL"

        try:
            variable = ele['variableSet']['name']
        except KeyError:
            variable = "NULL"

        try:
            fmcEvents = ele['sendEventsToFMC']
        except KeyError:
            fmcEvents = "NULL"

        try:
            begin = ele['logBegin']
        except KeyError:
            begin = "NULL"

        try:
            end = ele['logEnd']
        except KeyError:
            end = "NULL"

        try:

```

```

        syslog = ele['enableSyslog']
except KeyError:
    syslog = "NULL"

enable = ele['enabled']

if ("timeRangeObjects" in ele.keys()):
    for local in ele['timeRangeObjects']:
        trObj = local['name']
else:
    trObj = "NULL"

#resetamos array de zonas
sZones = []
dZones = []
try:
    temp = ele['sourceZones']
    for key,value in temp.items():
        for local in value:
            sZones.append(local['name'])
except KeyError:
    sZones.append("Any")

try:
    temp = ele['destinationZones']
    for key,value in temp.items():
        for local in value:
            dZones.append(local['name'])
except KeyError:
    dZones.append("Any")

#resetamos array de redes
sNetwork = []
dNetwork = []

try:
    temp = ele['sourceNetworks']
    for key, value in temp.items():
        if ('literals' in key):
            for local in value:
                sNetwork.append(local['value'])
        if ('objects' in key):
            for local in value:
                sNetwork.append(local['name'])
except KeyError:
    sNetwork.append("Any")

try:
    temp = ele['destinationNetworks']
    for key, value in temp.items():
        if ('literals' in key):
            for local in value:
                dNetwork.append(local['value'])

```

```

        if ('objects' in key):
            for local in value:
                dNetwork.append(local['name'])
except KeyError:
    dNetwork.append("Any")

#resetamos array de puertos
sPort = []
dPort = []

try:
    temp = ele['sourcePorts']
    for key, value in temp.items():
        for local in value:
            if ('PortLiteral' in local['type']):
                sPort.append(local['type'])
            else:
                sPort.append(local['name'])
except KeyError:
    sPort.append("Any")

try:
    temp = ele['destinationPorts']
    for key, value in temp.items():
        for local in value:
            if ('PortLiteral' in local['type']):
                dPort.append(local['port'])
            else:
                dPort.append(local['name'])
except KeyError:
    dPort.append("Any")

#resetamos array de App/URL

app = []
url = []

try:
    temp = (ele['urls'])
    for key,value in temp.items():

        if ("urlCategoriesWithReputation" in key):
            for local in value:
                url.append(local['category']['name'])
        if ("literals" in key):
            for local in value:
                url.append(local['url'])
        if ("objects" in key):
            for local in value:
                url.append(local['name'])
except KeyError:
    url.append("Any")

try:

```

```

        temp = ele['applications']
        for key, value in temp.items():
            for local in value:
                if ("tags" in local.keys()):
                    item = local['tags']
                    for ite in item:
                        app.append(ite['name'])
                if ("Application" in local['type']):
                    app.append(local['name'])
    except KeyError:
        app.append("Any")

#VLAN Tags, por si alguien lo usa ^^

vlan = []

try:
    temp = ele['vlanTags']
    for key, value in temp.items():
        for local in value:
            if ('VlanTagLiteral' in local['type']):
                vlan.append(local['startTag'])
            else:
                vlan.append(local['name'])
except KeyError:
    vlan.append("Any")

#comentarios

comment = []

try:
    temp = ele['commentHistoryList']
    comment.append(temp)
except KeyError:
    comment.append("Not Applicable")

# Cabecera

    policyFile.write("{0}, {1}, {2}, {3}, {4}, {5}, {6}, {7},
{8}, {9}, {10}, {11}, {12}, {13}, {14}, {15}, {16}, {17}, {18},
{19}, {20}, {21}\n".format(contador,ele['name'],';'.join(sZones),
';'.join(dZones),';'.join(sNetwork),
';'.join(dNetwork),';'.join(sPort), ';'.join(dPort), ';'.join(url),
';'.join(app), vlan, ele['action'],enable, ips, variable, file,
begin, end, fmcEvents, syslog, trObj, comment))

    policyFile.close()

def init():

    hostname = "192.168.1.1"

```

```
username = "myuser"  
password = "mypassword123!"
```

```
fmc = FMC(hostname=hostname, username=username,  
password=password, domain='Global')
```

```
acp = "11111111-XXXX-YYY-0000-000000000000"
```

```
exportaCSV(fmc, acp)
```

```
init()
```