# UDS Enterprise

## Wake on LAN for physical machines with UDS Enterprise

www.udsenterprise.com

## Introduction

UDS Enterprise allows the publication of different types of services. Among them there is the remote connection to physical machines (either to a pool of machines or to a specific machine). Previously, when one of these physical machines was turned off and it was necessary to turn it on automatically, it was necessary to rely on third-party tools such as OpenGnsys, which allows sending Wake on LAN (WoL) to the computers.

Starting with the next release UDS Enterprise 3.5, this functionality will come built into the software natively. Until then, a patch on UDS servers to enable this new option will have to be applied.

This document explains how to download this patch and how to configure UDS Enterprise to send all the necessary information to one or more WoL servers so they are capable of starting physical computers at the request of users.

## Requirements

Below you will find the most important requirements that must be met in order to use WoL with UDS Enterprise:

- Have an active UDS Enterprise environment in version 3.0
- The physical machines that will receive the boot order must have WoL support
- The different elements in the network where the physical device is connected (routers, firewall, switches, etc ...) have to support the sending of the WoL
- You must have at least one WoL server (a sample WoL server is used in this document)

## Apply patch to UDS servers

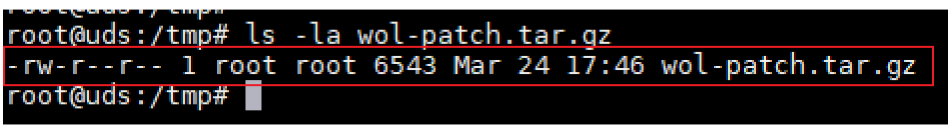
The first task to be performed will be to apply a patch to the UDS-Server (broker) machines. This patch will enable a new section in the service provider "**Static IP Machines Provider**" where you will indicate the connection data with the WoL server.

You need to apply this patch to all UDS-Server (broker) machines configured in the environment. To apply the patch you will perform the following tasks:

1. Download the patch from the following repository:

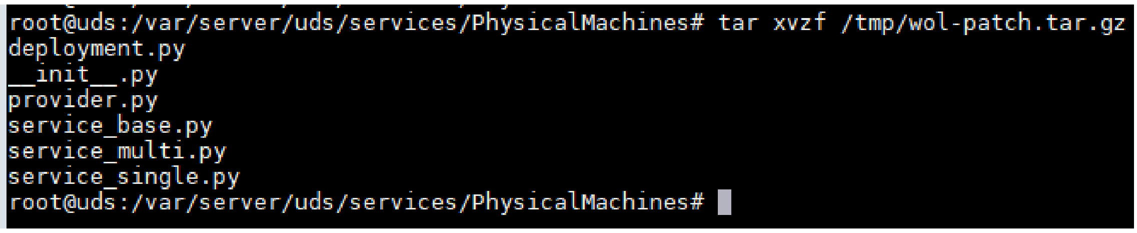
https://images.udsenterprise.com/files/UDSPatchs/UDS-3.0/WOL/wol-patch.tar.gz

2. Copy the downloaded wol-patch.tar.gz file to the UDS servers:



```
root@uds:/tmp# ls -la wol-patch.tar.gz
-rw-r--r-- 1 root root 6543 Mar 24 17:46 wol-patch.tar.gz
root@uds:/tmp#
```

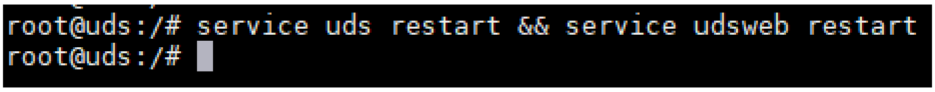
3. Unzip the contents of the previously downloaded patch in the path: **/var/server/uds/services/PhysicalMachines/**



```
root@uds:/var/server/uds/services/PhysicalMachines# tar xvzf /tmp/wol-patch.tar.gz
deployment.py
__init__.py
provider.py
service_base.py
service_multi.py
service_single.py
root@uds:/var/server/uds/services/PhysicalMachines#
```

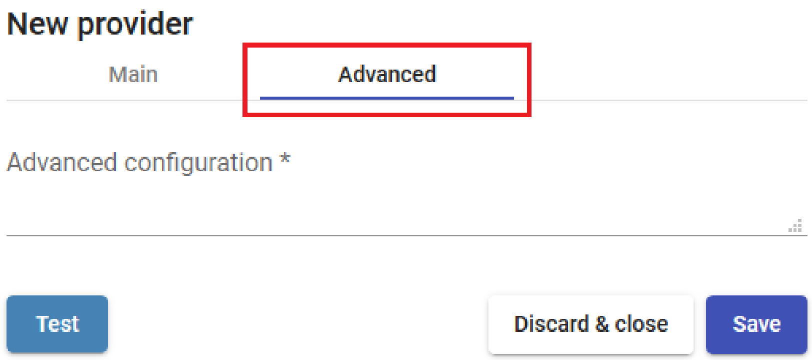
**NOTE:** If you want to save a copy of the old files, it is very important to take into account that these files cannot be hosted under the path **"/var/server/uds"** of the UDS server (even if the name, extension, etc. ...). It is recommended to copy them to a location outside that path or even off the server.

4. Restart the UDS servers or services: **uds** and **udsweb**



```
root@uds:/# service uds restart && service udsweb restart
root@uds:/#
```

Once the patch has been applied and the UDS servers or services have been restarted, when reloading the web administration you will see a new tab called "**Advanced**" within the "**Static IP Machines Provider**" service provider:



New provider

Advanced configuration *

## Wake on LAN server

The next element that you will have to configure will be the WoL server. It will be in charge of sending the "**Magic Packet**" to the physical computers so that they turn on. Depending on the complexity of your network and the different elements in it, you will have to configure a single server or several ones.

The sending of the WoL through the network can be interrupted by routers, firewalls, switches and other elements existing in the network. For this reason, it is advisable to place a WoL server for each network where the physical computers to be started are located. New network interfaces can also be added to the WoL server that connects to the different network segments where said device is located.

To facilitate the task of deploying this server, UDS provides an example Appliance with a series of very basic but sufficient resources for this task.

The server can be downloaded from the following repository:

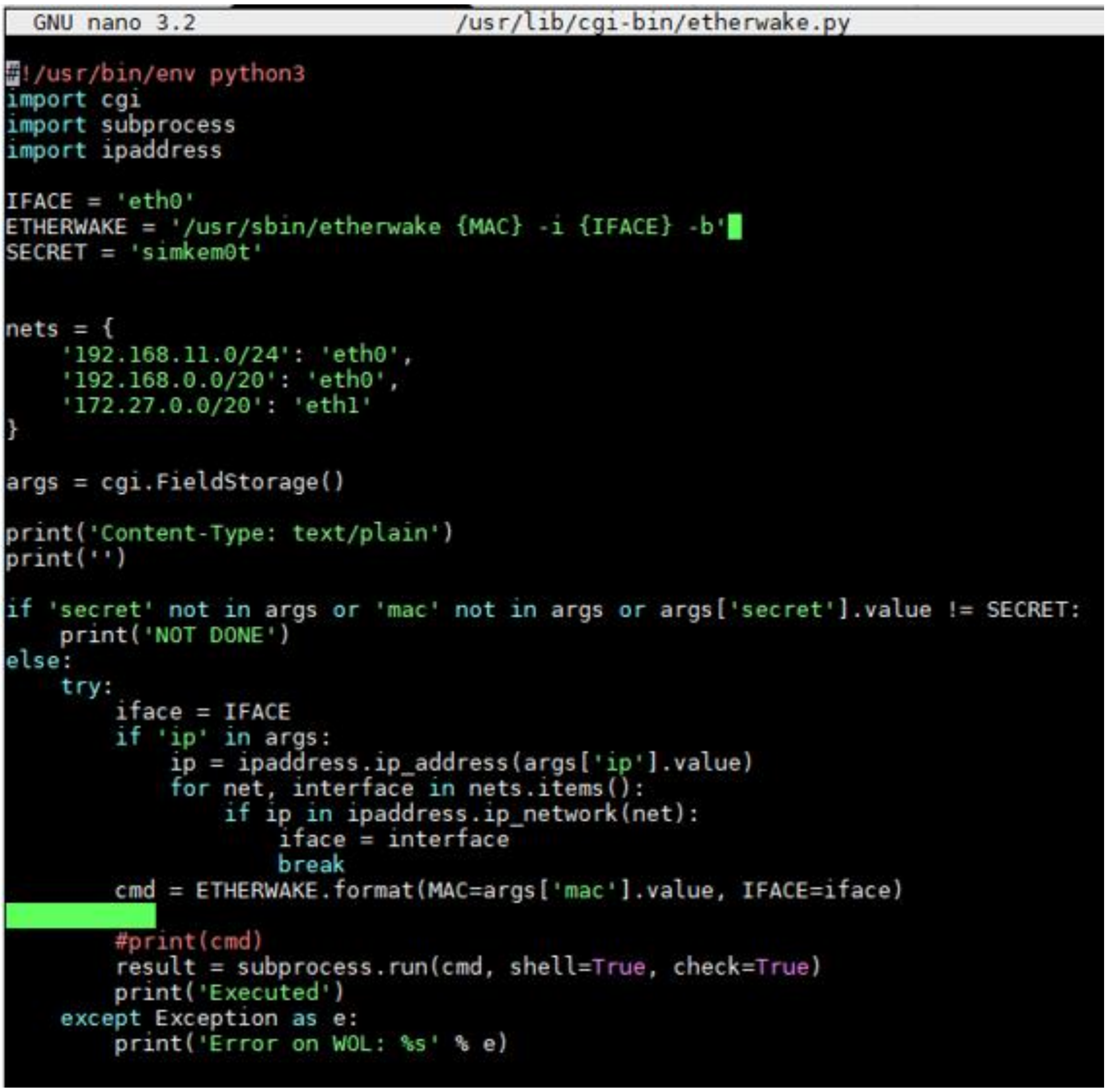
https://images.udsenterprise.com/files/UDSPatchs/UDS-3.0/WOL/Debian-WoL.ova

Server data:

- OS: Debian 10
- vCPUs: 1
- vRAM: 512 MB
- Disk: 8 GB
- User: root
- Password: uds

**NOTE**: The server is provided as a Virtual Appliance in **.ova** format. If it is necessary to convert it to a different format. To use it in other virtualization platforms, conversion tools such as **qemu-img** can be used (you can convert the disk in the **.vmdk** format extracted from the **.ova** into any other format).
The WoL server provided by the UDS Enterprise team includes the "**etherwake**" software, in charge of turning on the devices once its MAC address has been indicated.

Inside the server, in the path: **/usr/lib/cgi-bin/** you can find the **etherwake.py** script configured, as an example, as follows:



```
  GNU nano 3.2                    /usr/lib/cgi-bin/etherwake.py

#!/usr/bin/env python3
import cgi
import subprocess
import ipaddress

IFACE = 'eth0'
ETHERWAKE = '/usr/sbin/etherwake {MAC} -i {IFACE} -b'
SECRET = 'simkem0t'


nets = {
    '192.168.11.0/24': 'eth0',
    '192.168.0.0/20': 'eth0',
    '172.27.0.0/20': 'eth1'
}

args = cgi.FieldStorage()

print('Content-Type: text/plain')
print('')

if 'secret' not in args or 'mac' not in args or args['secret'].value != SECRET:
    print('NOT DONE')
else:
    try:
        iface = IFACE
        if 'ip' in args:
            ip = ipaddress.ip_address(args['ip'].value)
            for net, interface in nets.items():
                if ip in ipaddress.ip_network(net):
                    iface = interface
                    break
        cmd = ETHERWAKE.format(MAC=args['mac'].value, IFACE=iface)

        #print(cmd)
        result = subprocess.run(cmd, shell=True, check=True)
        print('Executed')
    except Exception as e:
        print('Error on WOL: %s' % e)
```

In the script you must take into account the following elements:

- Section "**SECRET**": You must indicate a string of alphanumeric characters to provide greater security to the connection between the UDS server and the WoL server. You will also use this same chain when you configure the WoL server from the UDS Enterprise administration. It is basically a security parameter, which prevents anyone who knows the WoL server from making unauthorized requests.

- Section **"nets**": Here you must indicate through which of the virtual network interfaces you will send the **"Magic Packet**" of the WoL. The WoL server provided will always use the nomenclature "**ethX**" (eth0, eth1, eth2, etc ...).

  If you have decided to place a WoL server for each of the physical devices networks, you will only use the **"eth0**" interface. You must indicate the subnet to which the devices belongs.

If on the WoL server there are different network interfaces (to use the same server for several subnets), you must indicate the subnet and the interface (in the example script it is indicated that on all physical computers that belong to the subnet: 192.168.11.0/24 and to the 192.168.0.0/20 subnet, the WoL is sent through the **"eth0"** interface. All the computers that belong to the 172.27.0.0/20 subnet will go through the **"eth1"** interface). It is possible to add as many networks (in CIDR format) and interfaces as necessary.

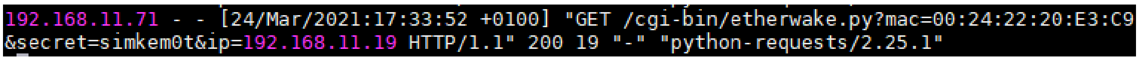
- "**IFACE**" section: If some of the IP addresses of the physical devices on which the WoL is to be sent are not defined in the "nets" section, the network interface indicated in this variable will be used. For example, if you want to send a WoL to 172.26.0.5 with the provided script, since your network is not defined in the "**nets**" section, it will go through the "**eth0**" interface.

If you want to manually check that the WoL servers work correctly on a specific computer, you can use the following command:

```
/usr/sbin/etherwake [MAC] -i [INTEFACE] -b
```

If the network connections are correct, and the machine supports WoL, it should boot up by running this command.

If you want to check if the device defined from the UDS broker is receiving the WoL and what data is being sent, you can access the following device log file to verify it: **/var/log/nginx/access.log**



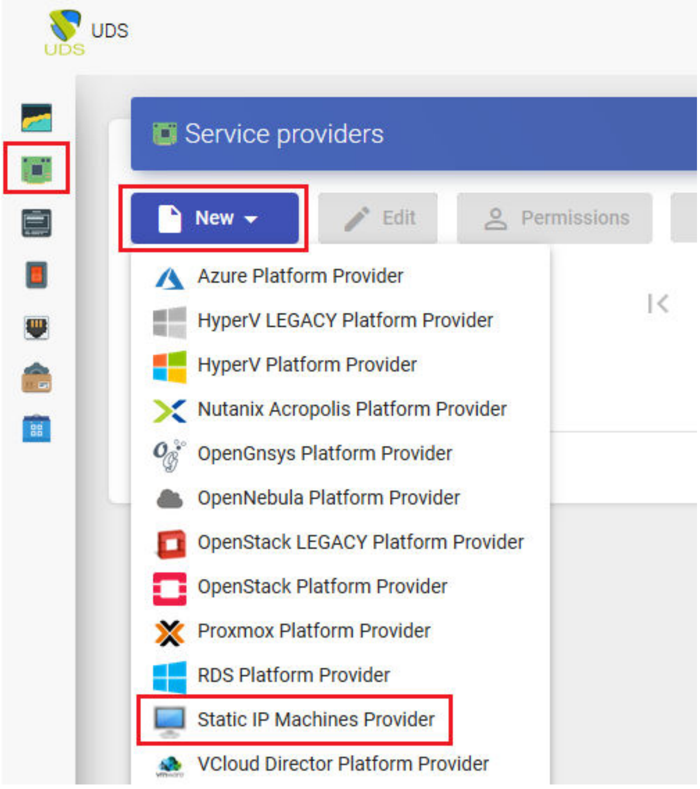
```
192.168.11.71 - - [24/Mar/2021:17:33:52 +0100] "GET /cgi-bin/etherwake.py?mac=00:24:22:20:E3:C9
&secret=simkem0t&ip=192.168.11.19 HTTP/1.1" 200 19 "-" "python-requests/2.25.1"
```

The first IP address will correspond to the server from where the WOL request is sent, usually the UDS server itself (in the example: 192.168.11.71). Then the execution of the **etherwake** script described above is indicated, followed by the MAC address of the device, the string **"SECRET"** and the IP address of the physical device to start (in the example: 192.168.11.19). The MAC and IP data will be those indicated in the UDS Enterprise administration.

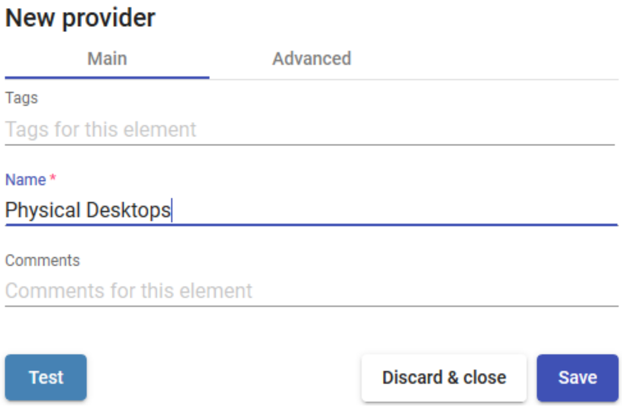
## UDS Enterprise Administration

You must have the MAC data and IP address (or DNS name) of the physical devices to which you want to send the WoL. Once you have these data, you must indicate them in the UDS Enterprise administration.

Access the UDS Enterprise administration (with a user with permissions) and go to the **"Services**" section. There you need to register a new **"Service provider**" of the **"Static IP Machine"** type:


UDS
Service providers
New
Edit
Permissions
Azure Platform Provider
HyperV LEGACY Platform Provider
HyperV Platform Provider
Nutanix Acropolis Platform Provider
OpenGnsys Platform Provider
OpenNebula Platform Provider
OpenStack LEGACY Platform Provider
OpenStack Platform Provider
Proxmox Platform Provider
RDS Platform Provider
Static IP Machines Provider
VCloud Director Platform Provider

Indicate a descriptive name for the provider's name:


New provider

Main | Advanced

Tags: Tags for this element

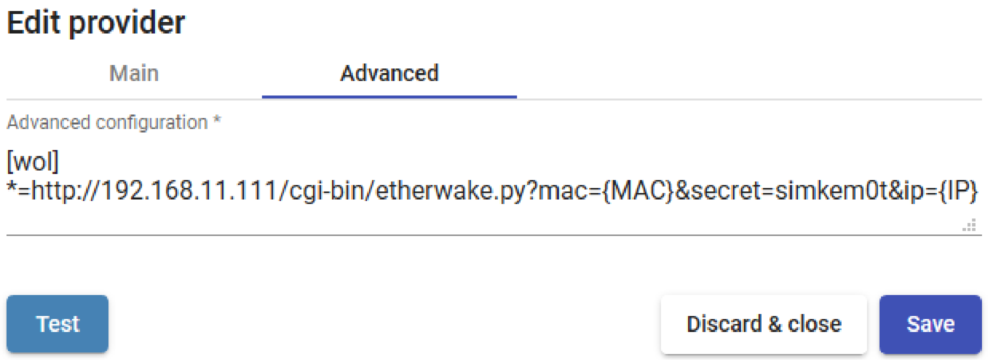
Name *: Physical Desktops

Comments: Comments for this element

In the **"Advanced"** tab you must indicate the option **[wol]**. Below you will add the rules for sending WoL to the servers configured for this purpose.

Example:

```
[wol]
* =http: //xxx.xxx.xxx.xxx/cgi-bin/etherwake.py?mac={MAC}&secret=xxxxxx&ip={IP}
```

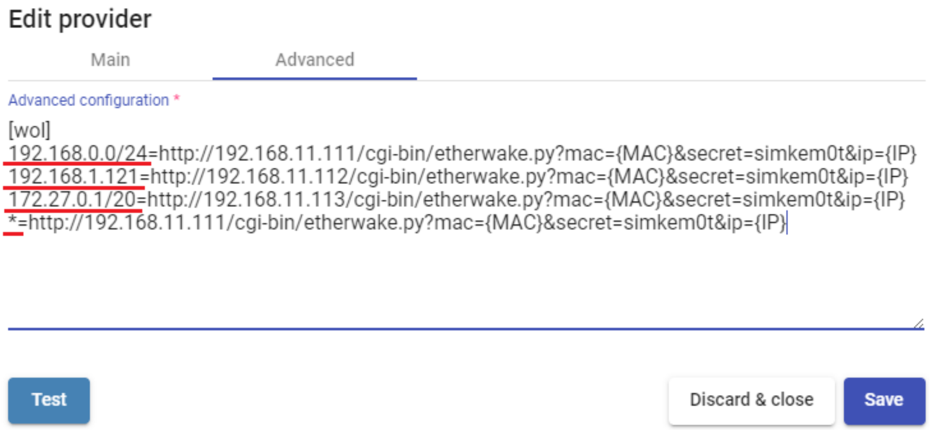


```
[wol]
*=http://192.168.11.111/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
```

The format of the rules will be as follows:

1. First you will indicate the subnet, network range or IP to which the physical devices will belong. If you introduce an asterisk, they will all be included.

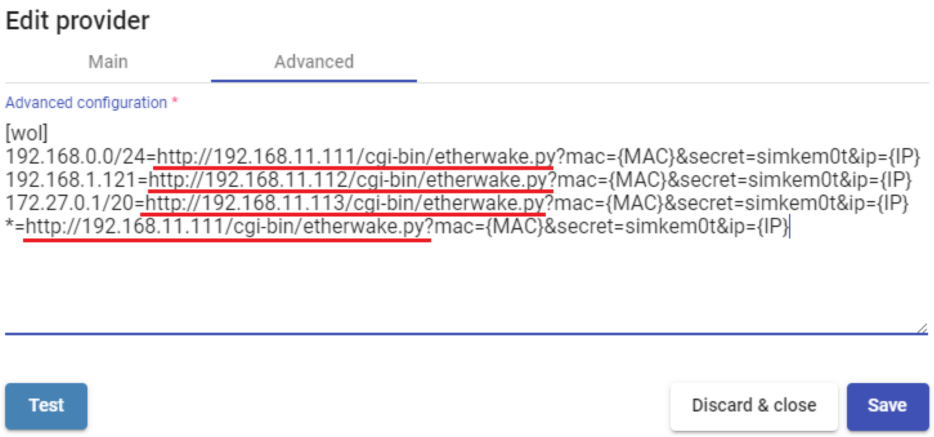
   In the following example we can check the following:
   - All computers belonging to the 192.168.0.0/24 subnet will be sent to the WoL server 192.168.11.111
   - The device with IP 192.168.1.121 will be sent to the WoL server 192.168.11.112
   - All computers belonging to the 172.27.0 / 20 subnet will be sent to the WoL server 192.168.11.113
   - The rest of the computers, that do not belong to the subnets or IP indicated above, will be sent to the WoL server 192.168.11.111



```
[wol]
192.168.0.0/24=http://192.168.11.111/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
192.168.1.121=http://192.168.11.112/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
172.27.0.1/20=http://192.168.11.113/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
*=http://192.168.11.111/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
```

2. Once the networks to which the physical devices belong have been defined, you will indicate the connection data with the WoL server that will send the **“Magic Packet**”.

   In the following example, you can see a configuration so that depending on the IP to which the physical device belongs, it is sent to a specific WoL server.

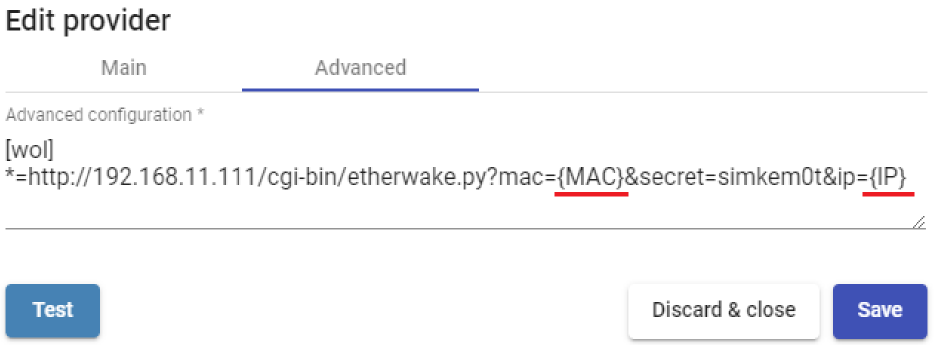


Edit provider

Advanced configuration *

```
[wol]
192.168.0.0/24=http://192.168.11.111/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
192.168.1.121=http://192.168.11.112/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
172.27.0.1/20=http://192.168.11.113/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
*=http://192.168.11.111/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
```

3. In all WoL requests to the servers, you will have to send the MAC address and the IP address. UDS Enterprise collects this data automatically when the device is defined in the **“Static Multiple IP”** base service and it is sent with the variables: **{MAC}** and **{IP}.**

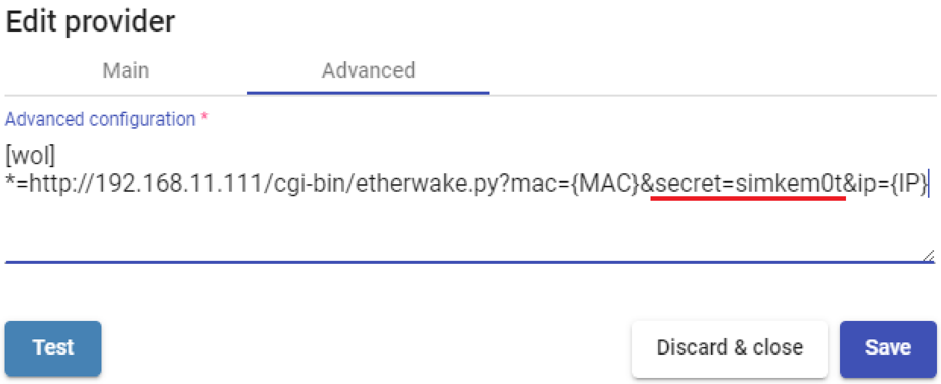


Edit provider

Advanced configuration *

```
[wol]
*=http://192.168.11.111/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
```

4. Finally you must indicate the **"SECRET"**, which will be a character string defined in the script **(etherwake.py)** of the WoL server.

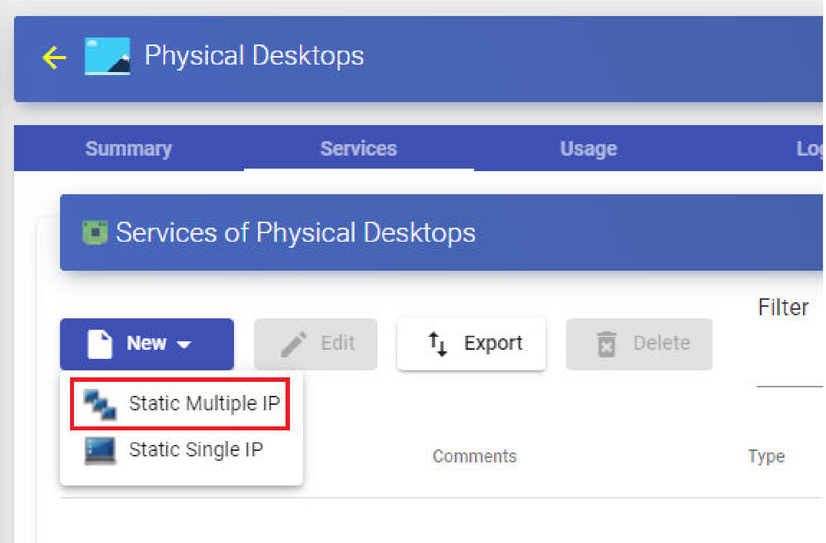


Edit provider

Advanced configuration *

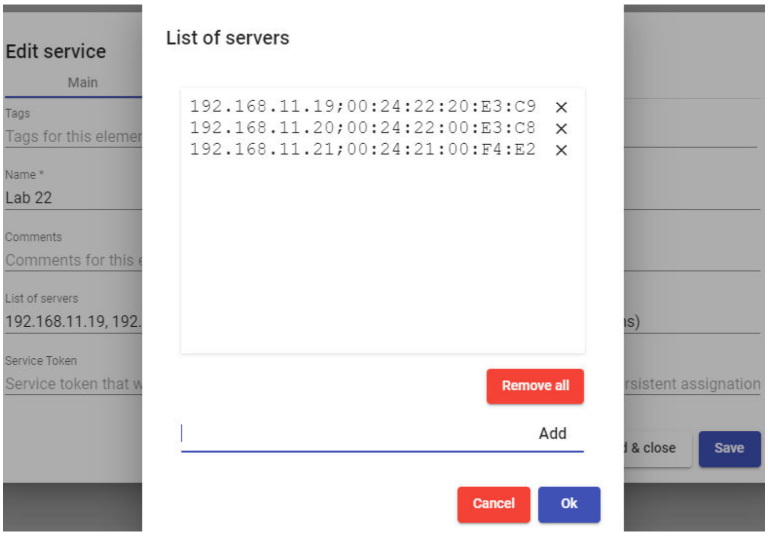
```
[wol]
*=http://192.168.11.111/cgi-bin/etherwake.py?mac={MAC}&secret=simkem0t&ip={IP}
```

Once the service provider has been created with the WoL configuration in the **“Advanced”** tab, access it and create a new base service of the **“Static Multiple IP**” type:



In this type of base service, you must indicate an identifying name for the service. In the "**List of Servers**" section, you will indicate the IP addresses or names of the physical devices to be used. Next you will introduce the MAC address of the device card, separated by a semicolon. The format would be:

*IP _Adress***;***MAC _Address*



Tags
Tags for this elemer

Name *
Lab 22

Comments
Comments for this e

List of servers
192.168.11.19, 192.

Service Token
Service token that w

List of servers

```
192.168.11.19;00:24:22:20:E3:C9 ×
192.168.11.20;00:24:22:00:E3:C8 ×
192.168.11.21;00:24:21:00:F4:E2 ×
```

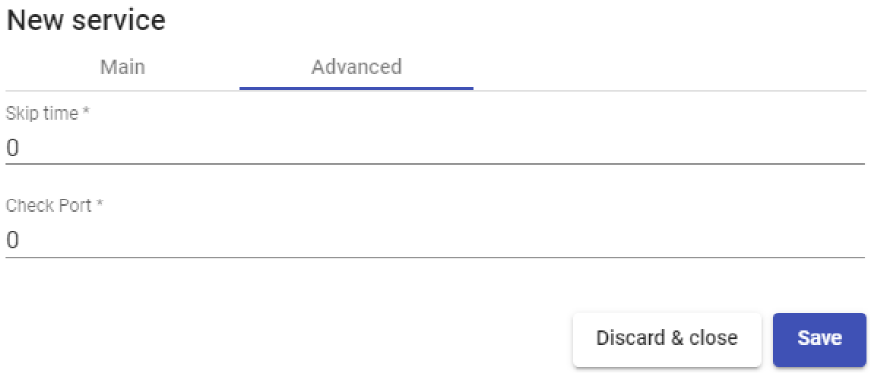
rsistent assignation

If you want UDS Enterprise to detect user logins and logoffs (essential if we need the machines to be released and available to other users), you must install the **"unmanaged"** actor and you can use the **"Service Token"** section. If what you need is that a machine is always assigned to a specific user, you will not use the **"Service Token".**

**NOTE**: It is recommended to use IP addresses instead of names. It is also possible to use the DNS name of the computers as long as there is name resolution from both the UDS servers and from the connecting client computers that execute direct connections. If you use **"Service Token"**, it is not possible to use names. Only the use of IP addresses is supported.

**NOTE2**: It is possible to combine devices with and without a MAC address. Devices that do not include the address will not be sent WoL.

In the **"Advanced"** tab, it is essential that the connection is not checked (that is, set **"Check Port"** to zero), since the device will be off. It is also not possible to use the "**Skip time**" value.



New service

Main | Advanced

Skip time *
0

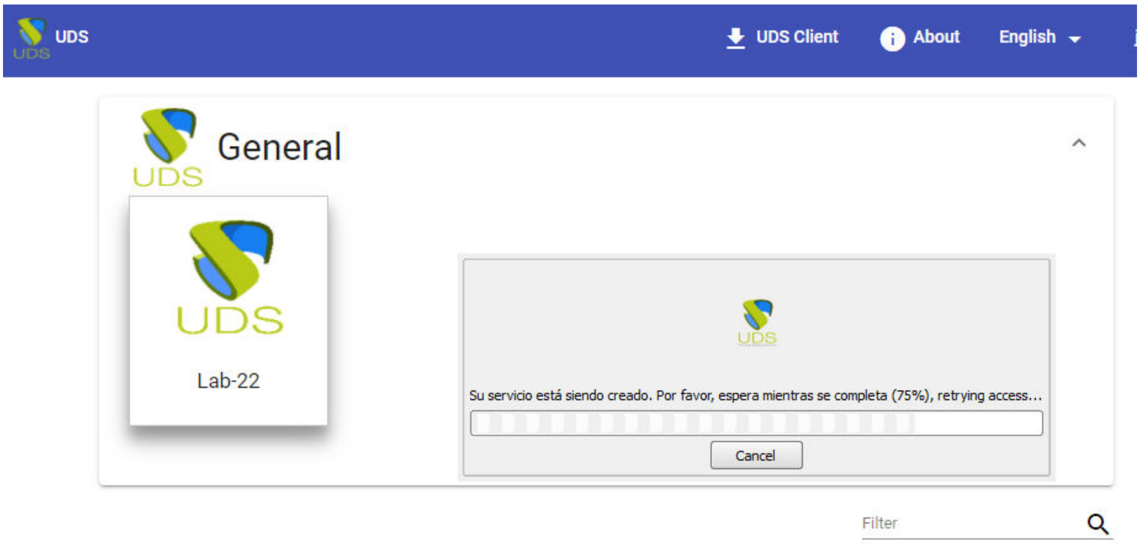
Check Port *
0

Once the **"Static Multiple IP"** type base service has been created and the devices to be used indicated, you will be able to create the **"Service Pool"** for user access.

## Machine access

Once all the necessary configurations have been made in the elements described above and the service pool has been created with at least one group of users and one transport, you will be able to access your physical devices.

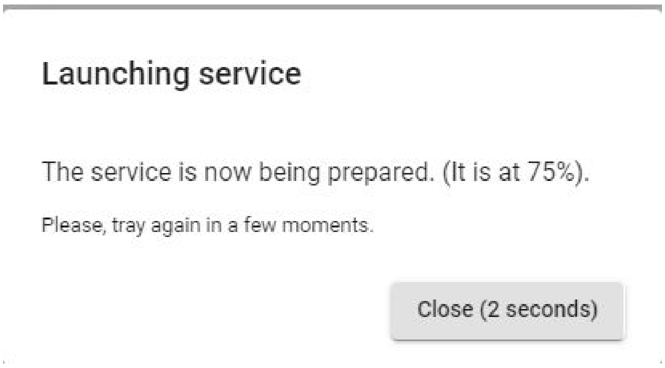
When you access, the team will have to start. While starting, the following notice will appear in the RDP connection box:



Su servicio está siendo creado. Por favor, espera mientras se completa (75%), retrying access...

Once the computer is started, UDS Enterprise will automatically connect to it (the time to complete the connection will depend on how long it takes to start the physical computer and to enable the RDP service).

If you are using an HTML5 type connection, the system will show this warning:


Launching service

The service is now being prepared. (It is at 75%).

Please, tray again in a few moments.

You will have to test the connection after a few seconds, until the device is completely started. (The HTML5 connection type does not wait for the machine to start, you have to access the service for the first time to start the machine. When it is started, you have to access the service again to start the connection).

# Professional services and support

Virtual Cable sells UDS Enterprise through a subscription model according to the number of users, including support and updates.

In addition, Virtual Cable offers professional services to install and configure UDS Enterprise.

For more information visit www.udsenterprise.com or email us at info@udsenterprise.com