

Installation Medulla

Documentation d'installation du serveur Medulla en monostite via ansible.

- [Installation OS debian pour serveur Medulla](#)
- [Installer Medulla sur Linux](#)
- [Installation de l'agent Medulla](#)
- [Déploiement Agent par WinRM SSH](#)
- [Medulla interface](#)
- [Configuration DHCP / PXE](#)

Installation OS debian pour serveur Medulla

Pour suivre les prérequis et de l'installation serveur debian pour Medulla veuillez cliquez sur le lien ci-dessous:

[Installation OS debian pour serveur Medulla](#)

Installer Medulla sur Linux

Pré-requis :

Le script d'installation de Medulla supporte la distribution Debian (Bookworm).

Vous devez disposer d'une machine avec :

- **au moins 8 Go de RAM disponibles,**
- **au minimum 40 Go d'espace disque libre pour / et 200Go pour le /var.**

Vérifier que votre fichier **/etc/hosts** est bien configuré avec le fqdn complet.

Si par exemple la machine s'appelle medulla et le domaine s'appelle domaine.lan, le /etc/hosts doit contenir une ligne de ce genre:

```
127.0.1.1    medulla.domaine.lan    medulla
```

Vérifier que le fichier **/etc/resolv.conf** soit bien configuré et que la machine est bien connectée à internet.

Installation du serveur Medulla:

Pour installer le serveur Medulla : Utilisez un user avec des droits sudo pour exécuter le script et la playbook d'installation.

1- Transférer le script sur le serveur.

2- Mettre les droits d'exécution sur le fichier :

```
taper : chmod +x install_from_ansible.py
```

3- Lancer l'installation du serveur :

```
./install_from_ansible.py
```

Attendez que le processus d'installation se termine, un résumé affichera tous les mots de passe nécessaires (copiez-les dans un endroit sûr).

Installation de l'agent Medulla

L'agent Medulla est téléchargeable depuis

<http://medulla.domaine.lan/downloads/win/Medulla-Agent-windows-FULL-latest.exe>

L'agent Medulla peut être installé manuellement ou en silencieux:

```
"Medulla-Agent-windows-FULL-latest.exe /S"
```

Le processus d'installation continuera après la fin de l'installation, il installera toutes les dépendances.

Il se termine lorsque l'ordinateur apparaît dans Medulla en bleu (en ligne).

Nom de la Machine	Description	Système d'exploitation	Type
 demo-win-2		Microsoft Version d'évaluation de Windows 11 Entreprise	Standard PC (Q35 + ICH9, 2009)

Agent Medulla

Par défaut, nous fournissons un agent global pour Medulla, (comme explique sur ce poste).
La gestion des entités par poste se fait via GLPI.

Cependant, si vous souhaitez **bénéficier d'une assignation automatique des machines aux entités directement via l'agent**, il est également possible de **générer des agents spécifiques par entité**. Cette option n'est pas activée par défaut, mais nous pouvons vous accompagner pour la mettre en place si elle correspond à vos besoins. Pour voir les agents par

Entité : [Gestions des entités](#)

Déploiement Agent par WinRM SSH

Préparation des Machines

Machines Debian (Medulla Main et Medulla Relai)

- Assurez-vous que netcat est installé sur les machines Debian :

```
sudo apt update && sudo apt install netcat-openbsd
```

Machine Windows (Poste Client)

- Assurez-vous que PowerShell est configuré pour exécuter des scripts :

```
Set-ExecutionPolicy RemoteSigned -Scope CurrentUser
```

- Répondre [A] "Yes to all"

Scripts d'Écoute

Sur les Machines Debian (Medulla Main et Medulla Relai)

Utilisez le script `listen_ports_debian.sh` pour mettre en écoute les ports nécessaires.

Étapes :

1. **Téléchargez le script** `listen_ports_debian.sh` sur la machine.
2. **Rendez-le exécutable** (et convertir en unix si nécessaire) :

```
chmod +x listen_ports_debian.sh  
dos2unix listen_ports_debian.sh
```

1. **Exécutez le script :**

- Pour **Medulla Main** :

```
./listen_ports_debian.sh --medulla
```

- Pour **Medulla Relai** :

```
./listen_ports_debian.sh --relay
```

Sur la Machine Windows (Poste Client)

Utilisez le script `listen_ports_windows.ps1` pour mettre en écoute les ports nécessaires.

Étapes :

1. **Téléchargez le script** `listen_ports_windows.ps1` sur la machine.
2. **Exécutez le script :**

```
.\listen_ports_windows.ps1
```

Scripts de Test de Connexion

Sur les Machines Debian (Medulla Main, Medulla Relai et Poste Client)

Utilisez le script `medulla_connection_check.sh`, `medulla_relay_connection_check.sh` et `windows_connection_check.ps1` pour tester les connexions vers les machines.

Étapes :

1. **Téléchargez le script** les scripts sur les trois machines (Medulla Main, Relai et Poste Client).

2. **Rendez-le exécutable :**

```
3. chmod +x medulla_connection_check.sh
    chmod +x medulla_relay_connection_check.sh
    chmod +x listen_ports_debian.sh
```

4. Si nécessaire (erreur lors du lancement des scripts Debian), les convertir en unix :

```
5. dos2unix medulla_connection_check.sh
    dos2unix medulla_relay_connection_check.sh
    dos2unix listen_ports_debian.sh
```

6. **Exécutez le script :**

- Pour tester les connexions depuis **Medulla Main** vers le Medulla Relai :

```
• ./medulla_connection_check.sh --relay <IP_Medulla_Relai>
```

- Pour tester les connexions depuis **Medulla Main** vers un Poste Client :

```
• ./medulla_connection_check.sh --client <IP_Poste_Client>
```

- Pour tester les connexions depuis **Medulla Relai** vers Medulla Main :

```
• ./medulla_relay_connection_check.sh --medulla <IP_Medulla_Main>
```

- Pour tester les connexions depuis **Medulla Relai** vers un Poste Client :

```
• ./medulla_relay_connection_check.sh --client <IP_Poste_Client>
```

- Pour tester les connexions depuis **Poste Client** vers Medulla Main et Medulla Relai:

- ```
.\windows_connection_check.ps1 -Target <IP_Medulla_Main> -Mode pulse
.\windows_connection_check.ps1 -Target <IP_Medulla_Relai> -Mode relay
```

## Procédure Complète

### Mise en Écoute des Ports

#### Sur Medulla Main (Debian) :

```
./listen_port_debian.sh --medulla
```

#### Sur Medulla Relai (Debian) :

```
./listen_port_debian.sh --relay
```

#### Sur Poste Client (Windows) :

```
.\listen_ports_windows.ps1
```

### Test des Connexions

#### Depuis Medulla Main (Debian) :

```
./medulla_connection_check.sh --relay <IP_Medulla_Relai>
./medulla_connection_check.sh --client <IP_Poste_Client>
```

#### Depuis Medulla Relai (Debian) :

```
./medulla_relay_connection_check.sh --medulla <IP_Medulla_Main>
./medulla_relay_connection_check.sh --client <IP_Poste_Client>
```

#### Depuis le Poste Client (Windows) :

```
.\windows_connection_check.ps1 -Target <IP_Medulla_Main> -Mode pulse
.\windows_connection_check.ps1 -Target <IP_Medulla_Relai> -Mode relay
```

## Résolution des Problèmes

### Problèmes de Connexion

- **Vérifiez les adresses IP** : Assurez-vous que les adresses IP utilisées sont correctes.
- **Vérifiez les pare-feux** : Assurez-vous que les pare-feux sur les machines autorisent les connexions sur les ports nécessaires.

- **Vérifiez les services** : Assurez-vous que les services nécessaires sont en cours d'exécution.

### **Problèmes d'Écoute**

- **Vérifiez les ports en écoute** : Utilisez netstat -ano sur Windows ou ss -tulnp sur Debian pour vérifier que les ports sont bien en écoute.
- **Vérifiez les erreurs** : Consultez les messages d'erreur dans les scripts pour identifier les problèmes.

# Medulla interface

Pour accéder à la console Medulla, ouvrir un navigateur web avec l'adresse suivante :

<http://medulla.domaine.lan/mmc>

ou

<http://ip-serveur/mmc/>

Pour Découvrir Medulla et comment prendre en main la solution :

[Medulla Documentation](#)

# Configuration DHCP / PXE

## Enable UEFI boot with PXE

### DHCP server :

DHCP server needs special option to enable PXE boot,

The DHCP options are :

- Option 66
- Option 67

## Windows DHCP server configuration

First of all, the DHCP server needs to find out which type of computer is requesting to assign PXE server and the right file to boot.

To sort computer vendor classes must be defined.

### Vendor Classes

Vendor Classes as Detection Method is used to determine how devices are requesting a boot image from the DHCP server.

- Open the DHCP Console and expand the IPv4 Node
- Right-Click on 'IPv4 Node' and select 'Define Vendor Classes'
- Click 'Add'
- Create the UEFI 64-Bit Vendor class first by entering the following information
- Enter the following information for the respective fields:
  - **DisplayName:** PXEClient (UEFI x64)
  - **Description:** PXEClient:Arch:00007
  - **ASCII:** PXEClient:Arch:00007
- Click 'OK'
- Click 'Add'
- **DisplayName:** PXEClient (BIOS x86 & x64)
- **Description:** PXEClient:Arch:00000
- **ASCII:** PXEClient:Arch:00000
- Click 'OK'

## Creating Custom DHCP Policies

### BIOS 32-Bit & 64-Bit DHCP Policy

- Right-Click 'Policies' and click 'New Policy'
- Give the policy a friendly name that coincides with the your vendor class naming scheme:
  - **PolicyName:** PXEClient (BIOS x86 & x64)
  - **Description:** Delivers the correct bootfile for BIOS machines
- Click 'Next'
- On the 'Configure Conditions for the policy' page click 'add'
- Select the 'Value' drop-down box and select the **PXEClient (BIOS x86 & x64)** vendor class that you created in previous steps
- Ensure that you check the box '**Append wildcard(\*)**'
- Select 'Add'
- Select 'Ok'
- Click 'Next'
- If you want the policy to affect only a specific range within your scope configure it, otherwise select no and click 'next'
- On the Configure settings for the policy page ensure that 'DHCP Standard Options' is selected from the drop down box
- Configure the following scope options:
  - **066:** *IP Address of Medulla*
  - **067:** *bootloader/undionly.kpxe*
- Click 'Next'
- **On the Summary page click 'Finish'**

## UEFI DHCP Policy

- Right-Click 'Policies' and click 'New Policy'
- Give the policy a friendly name that coincides with the your vendor class naming scheme:
  - **PolicyName:** PXEClient (UEFI)
  - **Description:** Delivers the correct bootfile for (UEFI)
- Click 'Next'
- On the 'Configure Conditions for the policy' page click 'add'
- Select the 'Value' drop-down box and select the **PXEClient (UEFI)** vendor class that you created in previous steps
- Ensure that you check the box '**Append wildcard(\*)**'
- Select 'Add'
- Select 'Ok'
- Click 'Next'
- If you want the policy to affect only a specific range within your scope configure it, otherwise select no and click 'next'
- On the Configure settings for the policy page ensure that 'DHCP Standard Options' is selected from the drop down box
- Configure the following scope options:
  - **066:** *IP Address of Medulla*
  - **067:** *bootloader-uefi64/ipxe.efi*
- Click 'Next'

- On the **Summary** page click 'Finish'

## Remove Default PXE Options

Ensure that you have removed the 067, 066, 060 options from the default scope options to ensure that the Policies take precedence otherwise you will end up with conflict. As long as you have configured everything correctly you should now have the ability to boot machines from BIOS or UEFI.

## Linux DHCP Server

```
PXE definitions
option space PXE;
option PXE.mtftp-ip code 1 = ip-address;
option PXE.mtftp-cport code 2 = unsigned integer 16;
option PXE.mtftp-sport code 3 = unsigned integer 16;
option PXE.mtftp-tmout code 4 = unsigned integer 8;
option PXE.mtftp-delay code 5 = unsigned integer 8;
option PXE.discovery-control code 6 = unsigned integer 8;
option PXE.discovery-mcast-addr code 7 = ip-address;
option arch code 93 = unsigned integer 16;
In initial DHCP DISCOVER packet, PXE client sets option 93 to its arch.
0000 == IA x86 PC (BIOS boot)
0006 == x86 EFI boot
0007 == x64 EFI boot

PXE boot following the PXE specs
class "PXE" {
match if substring(option vendor-class-identifier, 0, 9) = "PXEClient";
vendor-option-space PXE;
option PXE.mtftp-ip 0.0.0.0;
if option arch = 00:07 {
filename "bootloader-uefi64/ipxe.efi";
} else {
filename "/bootloader/undionly.kpxe";
}
}

Etherboot boot
class "Etherboot" {
match if substring (option vendor-class-identifier, 0, 11) = "Etherboot-5";
option vendor-encapsulated-options 3c:09:45:74:68:65:72:62:6f:6f:74:ff;
```

```
option vendor-class-identifier "Etherboot-5.0";
vendor-option-space PXE;
option PXE.mtftp-ip 0.0.0.0;
}

subnet ##MEDULLA_NETWORK## netmask ##MEDULLA_NETMASK## {
option broadcast-address ##MEDULLA_BCAST##; # broadcast address
option domain-name ##MEDULLA_DOMAIN##; # domain name
option domain-name-servers ##MEDULLA_DNS##; # dns servers
option routers ##MEDULLA_GW##; # default gateway

pool { # Only defined pool

uncomment the two following lines for PXE-only boot
#allow members of "PXE"; # PXE-only
#allow members of "Etherboot"; # PXE-only
range ##MEDULLA_START## ##MEDULLA_END##;
next-server ##MEDULLA_IP##;
}
}
```