Clustered Application Server - Ulteo

Contents

Ulteo Remote Application Server

Installation, configuration, patching & troubleshooting guide to the Ulteo-OVD services. Additional details of this software can be found on their website. Here are some useful resources.

1. ***Ulteo home*** - http://www.ulteo.com/home/
2. ***Ulteo OVD source code*** - http://www.ulteo.com/home/en/download/sourcecode
3. ***Additional OVD source code access*** - http://archive.ulteo.com/mirror/ovd/releases/sources/
4. ***Community forums*** - https://groups.google.com/forum/?fromgroups#!forum/ulteo-ovd-community-support

Original Guides

All guides can be found @ http://doc.ulteo.com/latest and are recommended prior to applying the patch associated with this documentation.

Current (2012-08-28)

The current operating environments for the Ulteo-OVD application service details

Development environment

Currently development efforts for implementation of the Ulteo-OVD application service are as follows:

1. *Session manager* - Administrative interface for the development session manager: Ulteo-OVD Administrative panel (https://ulteo-srv.scl.utah.edu/ovd/admin)
2. *Client area* - Client area for using the Ulteo-OVD service: Ulteo-OVD Client login (https://ulteo-srv.scl.utah.edu/ovd/)
3. *Usage metrics* - In order to weigh the usefullness of this service I have setup some

simple statistical monitoring software which can be found at the following location: Awstats (http://ulteo-srv.scl.utah.edu/awstats/awstats.pl)

Live environment

The current live environment for the Ulteo-OVD service is as follows:

1. *Session manager* - Administrative interface for the development session manager: Ulteo-OVD Administrative panel (https://bifrost.scl.utah.edu/ovd/admin)
2. *Client area* - Client area for using the Ulteo-OVD service: Ulteo-OVD Client login (https://bifrost.scl.utah.edu/ovd/)
3. *Usage metrics* - In order to weigh the usefullness of this service I have setup some simple statistical monitoring software which can be found at the following location: Awstats (http://bifrost.scl.utah.edu/awstats/awstats.pl)

Overview, Setup & Configuration

This section will provide a general overview of the various components that make up the Ulteo OVD software. The diagram below illustrates how the web client interfaces with the configured application server(s).

http://www.ulteo.com/main/images/uovd/arch2.png

Session manager

The Session Manager component handles the Administrative panel which is used to configure the Ulteo software.

It uses the following locations...

1. ***/etc/ulteo/sessionmanager*** - Here you can find the Apache virtual host configuration directives, the default administrative login for the Ulteo admin interface etc.
2. ***/usr/share/ulteo/sessionmanager*** - The web interface for the session manager. This folder contains the administrative interface as well as components the webclient uses for authentication & session management.
3. ***/var/log/ulteo/sessionmanager*** - The logs here are used within the administrative interface and can serve as a good source of troubleshooting

Network ports & services list

Being a complex application there are several TCP/UDP port requirements for remote application usage. The session manager port requirements are as follows:

1. *Apache* - TCP ports 80 & 443 (I would recommend disabling port 80 and requiring access through 443)
2. *MySQL* - TCP port 3306 (disabling of outside access is fine due to session manager & web client using localhost for access)

3. *LMSocialServer* - TCP port 1111 (This port is used for application server status updates and can be limited via local port filters to application servers only)

Configuration settings

The current configuration settings within the Ulteo session manager are as follows:

System settings

1. *System on maintenence mode* - no
2. *Administration console language* - autodetect
3. *Debug option list* - info, warning, error & critical
4. *Cache logs update every* - 30 seconds
5. *Cache logs expiry time* - a day
6. *Default user group* - ???
7. *Domain integration* - internal
8. *Maximum items per page* - 100
9. *Maximum number of running sessions* - 0
10. *Modules activation* - ApplicationDB, ApplicationsGroupDB, AuthMethod, ProfileDB, SessionManagement, SharedFolderDB, UserDB, UserGroupDB, UserGroupDBDynamic

Server settings

1. *Disable reverse FQDN checking* - yes
2. *Action when a server status is not ready anymore* - switch to maintenence
3. *Auto-recover server* - yes
4. *Remove orphan applications when the application server is deleted* - yes
5. *Auto register new servers* - yes
6. *Auto switch new servers to production mode* - yes
7. *When an Application Server have reached its "max sessions" limit, disable session launch on it ?* - yes

Domain integration settings

1. *Internal database profiles* - internal

Authentication settings

1. *AuthMethod* - CAS
2. *CAS Server URL* - https://go.utah.edu:443/cas

Session settings

1. *Default mode for session* - applications
2. *Default language for session* - english
3. *Default timeout for session* - 1 day
4. *User can launch a session even if some of his published applications are not available* - yes

5. *User can use a console in the session* - no
6. *Multimedia* - yes
7. *Redirect client drives* - full
8. *Redirect client printers* - yes
9. *RDP bpp* - 16
10. *Enhance user experience* - yes
11. *Enable user profiles* - yes
12. *Auto-create user profiles when non-existant* - yes
13. *Launch a session without a valid profile* - yes
14. *Enable shared folders* - yes
15. *Launch a session even when a shared folder's fileserver is missing* - yes
16. *Forceable paramaters by users* - none
17. *Enable Remote Desktop* - yes
18. *Sessions are persistent* - yes
19. *Show icons on user desktop* - yes
20. *Allow external applications in Desktop* - yes
21. *Desktop type* - any
22. *Servers which are allowed to start desktop* - empty
23. *Enable Remote Applications* - yes

Events settings

1. *Email address to send alerts to* - User definable
2. *Server status changed* - checked
3. *Session startup* - checked
4. *SQL failure* - checked

Web interface settings

1. *Display users list* - no
2. *Public Webservices access* - yes

Groups

Because we are using the CAS (common authentication service) a dynamic group must be configured to handle users coming from this service.

In order for this dynamic group configuration you must first enable the 'DynamicGroupDB' module. You can do this by this series of clicks...

1. *Login to the administration area*
2. *Select Configuration*
3. *System Settings*
4. *Modules Activation*
    1. *Check 'DynamicGroupDB' option*
5. *Save*

Now that the required module is enabled follow this series of clicks to create a dynamic group...

1. *Users*
2. *Users Groups*
    1. *Create new group*
    2. *Dynamic*
    3. *Enter a unique name*
    4. *Add a unique description*
    5. *Cached* = no
    6. *Validation type* = "at least one"
    7. *Login stats with* = "u"
3. *Save*

Publications

In order for any of our CAS authenticated users (members of our new dynamic group) to use any of the applications the Ulteo software provides you must first create a list of published applications. The following series of clicks will do this.

1. *Publication Wizard*
2. *Use usergroups*
3. *Select dynamic group you just created*
4. *Next*
5. *Create group with applications*
6. *Select any/all applications you wish to provide to this dynamic group*
7. *Next*
8. *Enter unique name*
9. *Enter unique description*
10. *Next*
11. *Confirm*

Web client component

The web client component is the access point that clients wishing to launch virtualized/remote applications will use. This component relies upon java applets once authentication has occured to load the requested piece of software. It can be found in /usr/share/ulteo/webclient.

It uses the following locations...

1. ***/etc/ulteo/webclient*** - The primary configuration for the webclient can be found here.
2. ***/usr/share/ulteo/webclient*** - The webclient application including the Java applets, ajaxplorer etc can be found in this location.

Current configuration changes

Default to portal or application mode

Force portal mode for clients edit /etc/ulteo/webclient/config.inc.php

Uncomment:

```
define('OPTION_FORCE_SESSION_MODE', 'applications');
```

Force the default session manager URI

You may wish to force the default session manager URL edit /etc/ulteo/webclient /config.inc.php

Uncomment & edit:

```
define('SESSIONMANAGER_HOST', '[FQDN of session manager]');
```

Application server(s)

Linux

The linux application server is used to provide the file system interface and mapping to local shares for the remote authenticated user. Below are details of the installed environement.

Network ports & services list

The linux application server & filesystem uses several processes to make up the whole. Included in the ulteo-ovd-subsystem processes are the following:

TCP

1. *Apache* - The apache webserver using TCP port 1113 (This port only needs to be accessible to & from the session manager)
2. *Python* - A customized python client socket is open on TCP port 1112 (This port also only needs to be accessible to & from the session manager)
3. *NetBIOS* - The netbios service initialized from the Samba service using TCP port 139 (This port is required for the file service for remote authenticated users)
4. *Xvnc* - The Xvnc service listening on TCP port 5910 (This also needs to be accessable for remote authenticated users)
5. *Xrdp* - The Xrdp service listening on TCP 3350 (This is only bound to the local loop back adapter or localhost address and does NOT need to be publicly accessible for remote authenticated users)
6. *Cups* - The cupsd service listening on TCP port 631 (This also is only bound to the local loop back adapter or localhost and does NOT need to be publicly accessible for remote authenticated users)
7. *Samba* - The SMB service is bound to TCP port 445 (This port only needs to be accessible from the configured application servers)
8. *RDP* - The RDP (Remote Desktop Protocol) is bound to TCP port 3389 (This needs to be accessible from remote authenticated users)

Service details

Here are the details of the various files installed with the Ulteo-OVD subsystem (filesystem & application server) on a linux host.

1. */etc/ulteo* - The configuration file location for the Ulteo-OVD subsystem
2. */var/log/ulteo* - The log files for the Ulteo-OVD subsystem application server
3. */opt/ulteo* - The chroot environment used for the file system services as well as the application services

Windows

The windows application server is used by remote authenticated users to launch applications.

Network ports & services list

Being a complex application there are several TCP/UDP port requirements for remote application usage. The applicaiton port requirements are as follows:

TCP

1. *epmap* - TCP port 135 (This should only need to be accessible from the configured application servers)
2. *microsoft-ds* - TCP port 445 (This should also only need to be accessible from the configured application servers)
3. *Python* - TCP port 1112 (Also only needs to be accessible from the configured application servers)
4. *RDP* - TCP port 3389 (This needs to be accessible from any authenticated user)

UDP

1. *microsoft-ds* - UDP port 445 (Accessible from configured application servers)
2. *isakmp* - UDP port 500 (Also only accessible from the configured application servers
3. *ipsec-msft* - UDP port 4500 (Also only accessible from the configured application servers
4. *netbios-ns* - UDP port 137 (Accessible from configured application servers
5. *netbios-dgm* - UDP port 138 (Also accessible from configured application servers

Patching (to provide CAS authentication)

As of this writing (2012-08) CAS authentication for the Ulteo-OVD software is broken. The phpCAS::Client performs a redirect to the CAS authentication service when no ST or PG ticket exists on the client. However due to the authentication form posting credentials to the sessionmanager which then generates an XML formatted query prior to performing this redirection header information does not work properly.

The steps following will upgrade the current phpCAS module and implement the proper redirection based on the Ulteo-OVD CAS enabled options within the Ulteo-OVD admin interface.

Latest patch

Here is the latest patch (https://raw.github.com/jas-/ulteo/master/ulteo-latest-CAS.patch) which will update the phpCAS client included with the latest version of the Ulteo Session Manager. Please note that you must have the 'DynamicGroupDB' module enabled and also have defined a group using the DynamicGroupDB module as listed above for the Session Manager configuration section.

```
%> wget https://raw.github.com/jas-/ulteo/master/ulteo-latest-CAS.patch
```

Make backup

You should first make a backup of the /usr/share/ulteo folder. This folder contains the session manager and the web client (if installed on the same web server).

```
%> cd /usr/share && tar zcvf ~/ulteo-backup.tgz ulteo/
```

Apply patch

In order to apply the patch to the latest Ulteo installation (v3.x) you must first remove the outdated phpCAS installation. This is why the backup in the previous step is crucial should something go wrong. To do this issue the following command.

```
%> rm -frv /usr/share/ulteo/sessionmanager/PEAR/CAS*
```

Next simply apply the patch using the following command.

```
%> cd /usr/share && patch -p0 < ~/2012-08-24.patch
```

Troubleshooting

Here are some general troubleshooting guidelines to the various components that make up the Ulteo-OVD service.

Session manager

Server status

Occasionally an application server status will be in a 'broken' state. Generally this

refers to the application server process is no longer sending status updates to the session manager.

When this type of proplem occurs a restart of the Ulteo-OVD application service must be restarted.

Broken windows application server

Here are some common problems encountered when using the Ulteo-OVD application server (v3.0.2) in a Windows 2003 server environement.

Not listed in session manager

If the Windows application server is not registering within the Ulteo-OVD session manager there are a couple of DNS errors that could be the cause of the problem.

1. *FQDN of session manager* - During the installation process you were prompted to enter a session manager hostname, if an IP address was entered you may experience problems with the application server not registering with the session manager
2. *DNS A record* - If the DNS A record of the session manager OR the application server is incorrect you may experience problems with the application server registering with the session manager

In order to resolve these problems the following solutions may be applied.

1. *Use FQDN* - Use of a FQDN (Fully Qualified Domain Name) during the installation is highly recommended.
2. *Static host entry* - Although not recommended a static route can be added to the "C:\Windows\System32\drivers\etc\hosts" file and would look like this example.

```
127.0.0.1    localhost
10.0.0.2    hostname.of.session.manager    hostname
```

3. *DNS A record* - The addition or modification of the DNS A record corresponding to your session manager hostname

Exceptional condition

Windows 2003 server error logs when handling exceptional conditions may return errors similar to the following in the event viewer. The error listed below is due to a problem with the XML formatted response from the session manager when recieving a status request. This error could be an indication of a man in the middle attack scenario because the application server is expecting an XML formatted query from the session manager.

```
The instance's SvcRun() method failed
Traceback (most recent call last):
  File "win32serviceutil.pyc", line 806, in SvcRun
  File "OVDWin32Service.pyc", line 95, in SvcDoRun
  File "ovd\SlaveServer.pyc", line 167, in loop_procedure
```

```
  File "ovd\SMRequestManager.pyc", line 169, in send_server_monitoring
  File "ovd\SMRequestManager.pyc", line 69, in get_response_xml
IOError: (9, 'Bad file descriptor')
%2: %3
```

The above error is caused by the following query from the session manager.

```
[608] content type: text/html
```

And usually results in errors similar to the following:

```
Windows saved user ULTEO-WIN2K3\OVDAdmin registry while an application or service was still using the registry d
by the user's registry has not been freed. The registry will be unloaded when it is no longer in use.

 This is often caused by services running as a user account, try configuring the services to run in either the L
```

Although this scenario is rare, mitigation of this problem in the future is to modify the Ulteo-OVD application server to use a LocalServer or NetworkService account as stated in the error. This is possible by using the 'services' administrative panel to modify the running user account. However, due to problems with the system account used to run the service errors in creating profiles and mapping SID to the user accounts will fail due to privilege errors because the specified account must be able to create users & their associated profiles.

*As of this writing (2012-08-27) the ulteo service must be run as the '*OVDAdmin*' user account (default user created during installation of the OVD Application server).*

To resolve this communication error between the Ulteo Application server and session manager the service must be stopped and restarted. You can use taskmanager or the administrative services managmement console to do this.

Broken linux application server

A linux application server serves dual roles. It first provides linux applications and it also provides file system drive & printer mapping to authenticated clients.

1. *Offline* - If the application & file server is not available using the Ulteo-OVD administrative interface the ulteo-ovd-subsystem must be restarted.
2. *No file browser* - If an authenticated user connects to the service and does not see a file browser it is due to the Ulteo-OVD SMBD service being down or that no Linux application & file server has been registered.
3. *Cannot save to desktop* - If an authenticated user cannot save to their desktop it is because their current OS username does not match the authenticated username provided to the Ulteo-OVD service or the necessary samba file service to WebDAV folder mapping did not take place

In most situations these problems can be resovled by simply restarting the Ulteo-OVD-subsystem (from a command line)

```
%> sudo -c '/etc/init.d/ulteo-ovd-subsystem restart'
```

Broken Windows application server

A windows application server provides remote application to authenticated clients using terminal services connections.

1. *Offline* - If the windows application service is shown as broken or offline using the Ulteo-OVD sessionmanager administrative interface, or clients are not able to access windows applications, the Ulteo-OVD-slaveservice may need to be restarted. Use the *Administrative Tools -> Services* MMC snap-in to stop and restart the service. If the service cannot be restarted use the *Task Manager* to stop any OVD services and restart.

Additional recommendations (hardening the service)

Because of the many components this is broken into sections each component such as the session manager or application server is broken down into the core services each provide.

Session manager

Here are some additional configuration options you may apply to the default session manager installation.

Web server

1. ***Use of ACL's*** *(Session manager administrative control panel)* - The use of an allowed/deny list should be used within the */etc/ulteo/sessionmanager/apache2-admin.conf* to limit administrative access. An example follows:

```
Alias /ovd/admin /usr/share/ulteo/sessionmanager/admin
<Directory /usr/share/ulteo/sessionmanager/admin>
    Options FollowSymLinks
    AllowOverride None
    Order allow,deny
    allow from 192.168.1.0/24
    allow from 10.0.1.0/24
    deny from all
    DirectoryIndex index.php
    php_admin_flag magic_quotes_gpc Off
</Directory>
```

2. ***Use of ACL's*** *(Session manager application server monitoring component)* - Additionally the use of ACL's for the webservices component to limit connections from anywhere other than valid application servers can be used. To do this you must modify the */etc/ulteo/sessionmanager/apache2-vhost-server.conf*. Here is an example:

```
NameVirtualHost *:1111
Listen 1111
```

```
<VirtualHost *:1111>
        RewriteEngine on

        RewriteCond %{REQUEST_URI} ^/(.+)/(.+)$
        RewriteRule . /%1_%2.php [L]

        DocumentRoot /usr/share/ulteo/sessionmanager/webservices
        <Directory /usr/share/ulteo/sessionmanager/webservices>
                Order deny,allow
                deny from all
                allow from 192.168.1.10 #Linux application/file server
                allow from 192.168.1.11 #Windows application server
                allow from 192.168.2.0/24 #Or if you use an entire subnet for your application servers
        </Directory>
</VirtualHost>
```

Keep in mind this modification should take place anytime a new linux or windows application server is added to the Ulteo-OVD service.

3. ***SSL certificate*** - The default installation uses a self-signed certificate which should be replaced with a valid signed certificate based on the hostname used for the Ulteo DNS A record. Once a certificate has been signed modify the */etc/ulteo /sessionmanager/apache2-vhost-ssl.conf* to reflect this change. An example follows:

```
SSLEngine on
SSLCertificateFile /path/to/valid/signed/certificate.cer
SSLCertificateKeyFile /path/to/valid/private/key/used/for/certificate/generation.key
```

And here is how to create the certificate request from a certificate authority:

```
%> openssl genrsa -des3 -out server.key 1024
%> openssl req -new -key server.key -out server.csr
```

Then send the server.csr to the certificate authority for signing.

4. ***Force SSL*** - By default Apache will bind to port 80, because authentication is involved it is wise to force redirects to the SSL/TLS protocol on port 443. To do this we will add a simple redirect to the '/etc/apache2/sites-available/default' virtual host declaration like so:

```
RewriteEngine on
ReWriteCond %{SERVER_PORT} !^443$
RewriteRule ^/(.*) https://%{HTTP_HOST}/$1 [NC,R,L]
```

Database

1. ***MySQL user*** - The default installation does not create and associate a user which can be used to access the MySQL database. This is ***strongly*** recommended and can be done with the examples show below:

```
%> mysql -u root -p -e 'CREATE USER "[dbUser]"@"localhost" IDENTIFIED BY "[dbPassword]"'
%> mysql -u root -p -e 'GRANT SELECT, INSERT, UPDATE, DELETE, REFERENCES, INDEX, CREATE TEMPORARY TABLES,
%> mysql -u root -p -e 'FLUSH PRIVILEGES'
```

Once you have created the user account, assigned a password, issued permissions & flushed the privilege table log into the session manager using the default administrative account, browse to *configuration -> database settings* and update the fields to reflect your new account. In order to prevent errors during initial

configuration you should use the root MySQL user account and configuration the administrative section of the Ulteo-OVD session manager.

PHP

The PHP interpeter can also be hardened with the assistance of the suhosin patch. To install simple run the following as a root user:

```
%> apt-get install php5-suhosin
```

Once it is installed it is wise to configure it. Below are some options to harden this feature providing the maximum protection for the PHP interpreter.

1. *Executor options* - The suhosin patch can be used to prevent things such as directory traversals, stack execution depths and white/black listing of specific PHP functions. Below are the 'minimum' options to be configured for this section.

   ```
   suhosin.executor.max_depth = 50
   suhosin.executor.include.max_traversal = 5
   suhosin.executor.disable_eval = on
   ```

   I also highly recommend disabling the /e modifier available within the PCRE (perl compatible regular expression library) as they contain remote execute of scripts. *However,* this option requires modification of the PHP source code within the Ulteo-OVD software to remove all instances of the /e modifier used in the preg_match() function.

   ```
   suhosin.executor.disable_emodifier = on
   ```

2. *Misc options* - Apache may cause segfaults due to the APC functionality reserving resources which the suhosin patch may request. If your apache installation is causing segfaults you may wish to enable the APC workaround like so:

   ```
   suhosin.apc_bug_workaround = on
   ```

3. *Transparent encryption options* - Additionally transparent encryption of PHP server side session's & the client side cookie is recommended. Below are examples:

   ```
   suhosin.session.encrypt = on
   suhosin.cookie.encrypt = on
   ```

Linux Application Server

The linux application server provides several services which you may additionally configure using the recommendations below.

Samba

Additional configuration settings for the Samba file server service (within the chroot environment) may be used. Below are some guides:

1. *File types* - Disabling specific file types using the 'veto files' configuration directive in the '/opt/ulteo/etc/samba/smb.conf' can be used like so (this example disables most common script types & executables):

```
veto files /$RECYCLE.BIN/*.cpp/*.exe/*.sh/*.php/*.pl/*.bat/
```

2. *System accounts* - Additionally preventing system account access can also be utilized like so:

```
invalid users = daemon, bin, sys, sync, games, man, lp, mail
```

3. *Security mode* - Currently the samba fileserver mode is set to 'share' which only requires a password be provided when mapping a share. According to documentation regarding security hardning of a Samba fileserver this is strongly discouraged. A stricter security mode should be set such as 'user' as in this example:

```
security = user
```

   This configuration does result the client no longer being able to save files directly to their own machine. A feature request will be sent to the current Ulteo-OVD maintainers regarding this.
4. *Interface security* - Force bind mode to only allowed interfaces as well as force socket connection mode. Example:

```
interfaces = eth0, lo
bind interfaces only = yes
socket options = TCP_NODELAY
```

Apache

Additional configuration settings may also be applied to the Apache web server service (also located within the chroot environment). Below are some recommendations:

1. *Hosts* - The apache webserver can be hardened by restricting access through the use of the 'hosts allow' directive limiting access only to the currently configured session manager when sending requests. Keep in mind if you decide to enable this only the clients added to this whitelist would be able to access the mapped WebDAV fileshare. Here is an example configuration for the '/opt/ulteo/usr/share /ulteo/ovd/slaveserver.conf':

```
NameVirtualHost *:1113
Listen 1113
<VirtualHost *:1113>
        DAVMinTimeout 600
        DAVDepthInfinity On

        Alias /ovd/fs /var/lib/ulteo/ovd/slaveserver/fs
        <Directory /var/lib/ulteo/ovd/slaveserver/fs>
```

```
                DAV on
                AuthNAme "WebDAV Storage"
                AuthType Basic
                AuthUserFile /var/spool/ulteo/ovd/fs.dav.passwd
                Require valid-user
                AllowOverride AuthConfig Limit

                Order allow,deny
                allow from 192.168.1.0/24
                allow from 10.0.1.0/24
                deny from all
        </Directory>
</VirtualHost>
```

This whitelisting feature might be beneficial to include within the administrative interface so a feature request will be filed with the Ulteo-OVD maintainers.

cups Service

1. *Use of ACL's* -The cups printing service may also be hardened with the use of access control lists. Much like ACL's in the Apache webservice limiting access to the cups service by allowed remote clients will aid in preventing unauthorized use. Below is an example configuration:

```
# Restrict access to the server...
<Location />
  Order allow,deny
  allow from 191.168.1.10 #Individual machine allowed access to the cups printing service
  allow from 192.168.2.0/24 #Entire subnet of allowed machines to the cups printing service
  deny from all
</Location>
```

As with many of the other services the Ulteo-OVD service implements the administration of these whitelists would prove beneficial if available through the current administrative interface.

Windows Application Server

The Windows Ulteo-OVD application server can also be further restricted, below are some available options for hardning the application server service on Windows (This guide was developed using Windows Server 2003).

1. *Terminal services* - Terminal services should the following options enabled. You can modify these settings using the *Administrative Tools -> Terminal Services Configuration* MMC snap-in.
    1. *Delete temporary folders on exit* = Yes
    2. *Use temporary folders per session* = Yes
    3. *Active Desktop* = Disable
    4. *Permission Compatibility* = Full Security
    5. *Restrict each user to one session* = Yes
2. *Windows firewall* - Limiting access to the services required by the Ulteo-OVD application server to a whitelist of allowed machines will help prevent unauthorized access. Configuration can modified using *Control Panel -> Windows Firewall*. Below are the recommended settings:
    1. *OVDWin[arch]Service.exe* - Edit the scope for this service to either use a

custom list of allowed machines or restrict to the current subnet of the server
2. *ulteo-ovd-slaveservice* - You can also edit the scope for this server to use a custom list of allowed machines or restrict to the current subnet of the server
3. *Remote Desktop* - This windows server can also be filtered by restricting access to a whitelist of allowed machines or the current subnet of the server.