1. Introduction and assumptions ................................................................. 3
2. Setup your network ............................................................................. 5
3. Install Awingu appliance ................................................................. 9
4. Prepare your windows back-end ...................................................... 11
5. The Awingu Installer ........................................................................ 13
6. Connect Awingu to your Active Directory .................................... 15
7. Enabling SSL ..................................................................................... 19
8. Multi Factor Authentication ............................................................ 21
9. Publish apps and desktops .............................................................. 23
10. Publish drives .................................................................................. 26
11. Customize the branding ................................................................. 27
Introduction and assumptions

This document is a summary of all the steps that are needed to setup a single-node, single-tenant Awingu appliance and integrate it into an existing Windows infrastructure. For more details or complex setups, please consult the Awingu Administration guide.

Next to this getting started guide there is also a set of short 3-5 minute instruction videos available to help you with the installation and configuration of Awingu. We recommend using these videos in combination with this getting started guide. Video's can be found via this link https://awingu.com/technicalvideos or by searching on YouTube for “recommend using”

What is Awingu?

Awingu is a (virtual) appliance that acts as a gateway and allows secure connections from any modern (HTML5) browser to existing company resources such as:

- Windows applications (via RDP)
- Windows & Linux full desktops (via RDP)
- Fileservers (via CIFS/Webdav)
- Intranet Web based applications (via reverse proxy)
- SaaS applications

On top of this Awingu adds an extra layer of security (MFA, audit, encryption, recording, ...) and collaboration (file & session sharing).

Awingu Architecture

There are 3 layers in the Awingu Architecture:

- The client layer which can be any web browser on any device on any operating systems that supports html5 web sockets. Practically this means that almost every browser (on the exception of Internet Explorer 9 and 10 and some very old versions of Firefox) should work.
- The Awingu Layer: this can be one or more Awingu virtual appliances.
- An existing windows based back-end: Windows AD (or LDAP) for user authentication, in case applications are published a Microsoft terminal server and in case files are published a CIFS.

The 3 layers are also completely isolated from each other and both the client and the back-end layer doesn't require any additional software installation. All connections from the client layer to the Awingu layer or done via http/https and from the Awingu layer to the Windows back-end layer via standard protocols like ldap, rdp and cifs. More details on the different network flows can be found in the “install the appliance and setup your network” chapter of this guide.

The Awingu (virtual) appliance is available for both

- On-premise installations (hyper-v, VMWare ESX, KVM, openstack,....)
Cloud installations (Azure, Amazon, Google, IBM).

The Awingu appliance can be downloaded from [https://download.awingu.com](https://download.awingu.com). We recommend to always download and install the latest version.

Awingu can be installed and configured without license. All functionality will work and up to two concurrent users will be able to login to Awingu. This allows you to test and setup without the need to purchase a license. Once you go into production an Awingu license needs to be bought.

**Assumptions**

For this guide we assume there is already an existing Windows which includes:

- A Windows Active Directory (No Azure AD Only setup)
- Windows applications servers (RDS / Terminal Servers) or VDI's (accessible via RDP)
- Fileserver supporting CIFS/SMB access

If this is not yet present Awingu offers on its support portal some guides that explain how to setup these windows components.

More details on this windows back-end can be found in the "prepare your windows back-end" chapter.

For this getting started manual we assume there is:

- Only a single Awingu instance. Please note that Awingu can be configured into a "clustered" setup for purposes of scaling-out or high availability. Details on scaling out Awingu can be found in the Awingu Administration guide
- Only one windows back-end. Please note that Awingu can be configured with multiple tenants and each tenant can also connect to a different Windows back-end. Details on multi-tenancy can be found in the Awingu Administration guide
Setup your network

Before we start the installation it is important to understand how the Awingu appliance can be integrated into the existing network.

As the Awingu appliance doesn't require any agents or browser plug-ins on the clients and only standard protocol access (so also no agents or additional software) towards the back-end, it can be easily installed into existing networks. Existing security equipment like firewalls, load balancers, SSL offloaders / reverse proxies can be reused.

Also this architecture allows the 3 layers (clients, Awingu and the windows back-end) to be upgraded independent from each other.

Important is that in any configuration Awingu needs to be accessible (from the outside world) via port 80 and/or 443. Other ports are not supported at this time.

In this getting started manual we will describe 3 possible network scenario's. For more advanced network setups please have a look at the full admin guide:

1. Awingu behind a simple port-forwarding firewall
2. Awingu behind a reverse proxy / load balancer that does SSL offloading
3. Awingu in a DMZ network

Awingu behind a simple port-forwarding firewall

This is a most simple scenario to deploy Awingu. From the firewall port 80 and or 443 is forwarded to Awingu which is in the same network as the company resources (AD, RDS, webServer and/or Fileserver). SSL offloading can be enabled on the Awingu appliance by using the build-in Awingu SSL offloader.

For the SSL offloading Awingu supports 2 methode: For this scenario to work you need to simply port-forward the incoming https traffic (tcp port 443) to Awingu. Awingu can do the SSL offloading on the appliance.

- Via standard certificates. In this case you need a certificate in the CRT / KEY format. Check the support portal on how to convert for example pfx certificates to CRT / KEY format
- By using letsencrypt. Letsencrypt is a free public service that generates certificates. By providing the external DNS name to Awingu the certificates are automatically requested and renewed every three months. To use letsencrypt also incomming http (tcp port 80) needs to be open.

For this setup to work there are as such 3 requirements:

1. Port 443 (https traffic) on the public ip address needs to be forwarded to Awingu. This assumes that no other service is already using this port. If this is the case an additional public ip address needs to be added to the firewall.
2. This setup can only be used for single node Awingu setups. When using multiple nodes you need to setup a loadbalancer / reverse proxy in front of Awingu (see next design)
3. A certificate is required in CRT/KEY format. In case you don’t have a certificate Awingu allows you to auto create certificates via the free letsencrypt service but in that case also port 80 (http) needs to be portforwarded.
Awingu behind a reverse proxy / load balancer that does SSL offloading

In this scenario there is an existing reverse proxy / load balancer / SSL-offloader. On that device there is a virtual host defined that forwards all traffic linked to a specific DNS record / host header to one or more Awingu appliance. In this scenario we also assume the reverse proxy / load balancer / SSL-offloader also takes care of the SSL-Offloading.

In this scenario Awingu is not directly accessible from the outside world and the reverse proxy / load balancer / ssl-offloader will terminate the incoming traffic first and then proxy it towards the Awingu appliance. The main advantages are that you can have multiple services that use https hosted on the same IP address and in case the device is capable of doing load balancing you can also use it in combination with a multi-node Awingu.

In this scenario it is important to configure the reverse proxy / load balancer / SSL-offloader correct so it doesn’t break the HTML5 web socket Awingu uses and also that it preserves the header information correct so that Awingu knows what the original IP-addresses were rather than getting always the IP-address of the reverse proxy / load balancer / SSL-offloader.

The admin manual contains all details and also on the support website you find some specifics for some vendor specific solutions like F5, Netscaler, NGinx, ... but in general follow headers need to be correctly set on the device:

<table>
<thead>
<tr>
<th>Header</th>
<th>Value</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connection</td>
<td>This value should be equal to “Upgrade”</td>
<td>Mandatory for getting web sockets to work</td>
</tr>
<tr>
<td>Upgrade</td>
<td>Should be equal to websocket in case of an websocket upgrade</td>
<td>Mandatory for getting web sockets to work</td>
</tr>
<tr>
<td>X-Forwarded-Proto</td>
<td>This value should be equal to “https”</td>
<td>Mandatory for getting web sockets to work</td>
</tr>
<tr>
<td>X-Real-IP</td>
<td>This should be the IP address of the requesting client</td>
<td>Recommended for security</td>
</tr>
<tr>
<td>X-Forwarded-For</td>
<td>This should be the IP address of the requesting client</td>
<td>Recommended for having correct audit</td>
</tr>
<tr>
<td>X-Forwarded-Host</td>
<td>This is the FQDN of the server name that was requested by the client</td>
<td>Recommended for having correct audit</td>
</tr>
<tr>
<td>Host</td>
<td>This is the FQDN of the server name that was requested by the client</td>
<td>Recommended for having correct audit</td>
</tr>
</tbody>
</table>

Awingu in a DMZ network

As Awingu only uses standard protocols and ports it can be easy also installed in a DMZ network.

Both scenarios as described above (portforwarding or reverse proxy) can be used to access the Awingu from the public network. In this chapter we describe all other network flows that are needed to get Awingu installed and working from in a DMZ or firewalled network:

- User authentication
Accessing resources like windows apps, storage, vdi's & web apps
Installation and other needed network traffic

When using a multi-node Awingu it is important that all nodes can freely communicate with each other. So all Awingu nodes of a multi-node cluster need to be in the same network with no firewalls between the individual nodes.

User Authentication network flows

Awingu uses LDAP or LDAPS to communicate with the AD or LDAP server. This communication is used to validate the user login & password and also fetch the security groups of the user. When password changes via Awingu are allowed also kerberos communication is needed and it’s mandatory to use LDAPS (tcp 636). Doing password changes via LDAP (tcp 389) communication will not work.

<table>
<thead>
<tr>
<th>Communication</th>
<th>From</th>
<th>To</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDAP: TCP port 389</td>
<td>Awingu</td>
<td>All AD or LDAP servers in the configuration</td>
<td>Only required when users need to be able to change password</td>
</tr>
<tr>
<td>LDAPS: TCP port 636</td>
<td>Awingu</td>
<td>All AD or LDAP servers in the configuration</td>
<td>Requires also certification on the AD / LDAP server**</td>
</tr>
<tr>
<td>KERBEROS: UDP port 88 and TCP port 88</td>
<td>Awingu Kerberos server</td>
<td>Only required when users need to be able to change password at next logon. The kerberos server should also have PTR (reverse DNS) and SRV records in place to locate the KDC server and define the protocol to use**</td>
<td></td>
</tr>
<tr>
<td>Radius: UDP port 1812</td>
<td>Awingu Radius Server</td>
<td>Only required when using an external, radius based, MFA server. Awingu also offers the possibility to integrate native with some 3rd party MFA providers like Du, SMSpasscode,....</td>
<td>More details on network flows can be found in the admin manual.**</td>
</tr>
</tbody>
</table>

Application & Storage network flows

Awingu uses RDP to connect to windows based applications and desktops. Besides network access no other access is needed. In case of using a Microsoft RDS broker Awingu not only need RDP access to the broker but also to all individual session hosts that are configured behind the broker.

For accessing storage Awingu supports both CIFS and WebDav.

Awingu can also publish web based applications. In case the websites are external to the organization the browser will directly connect to them and as such the traffic will not pass Awingu. Since Awingu 4.0 it is also possible to access internal websites. For this Awingu will act as a reverse proxy and as such needs to access the internal website.
## Communication

<table>
<thead>
<tr>
<th>Communication</th>
<th>From</th>
<th>To</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>RDP: TCP port 3389</td>
<td>Awingu</td>
<td>Session Hosts, VDI hosts and RDS brokers</td>
<td>Only needed when offering Awingu Drives via RDP</td>
</tr>
<tr>
<td>CIFS: TCP port 445</td>
<td>Awingu</td>
<td>All storage servers.</td>
<td>Only needed when offering Awingu Drives via CIFS</td>
</tr>
<tr>
<td>WebDav: TCP port 80/443</td>
<td>Awingu</td>
<td>All storage servers.</td>
<td>Only needed when offering Awingu Drives via WebDav</td>
</tr>
<tr>
<td>HTTP/HTTPS: TCP port 80/443</td>
<td>Awingu</td>
<td>Web servers</td>
<td>Only needed for internal web apps via the reverse proxy feature.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>In case the internal website runs on an other port than the standard ports 80 or 443 then also this non standard port need to be added to the firewall rules.**</td>
</tr>
</tbody>
</table>

**Other network flows**

During the Awingu installer we need to configure NTP, DNS and DB access. The installer itself runs on port 8080. After the installer finished this port is no longer needed.

In most simple cases the AD acts also as the DNS & NTP server but Awingu supports also more complex setups with different DNS servers per tenant.

To do this there are 2 levels of DNS:

- DNS servers can be configured on the global appliance level. These servers are configured during the installer.
- Each tenant can also be configured with optional DNS servers. These servers are configured when adding a tenant.

The optional DNS servers are used in the tenant to resolve the host names of the servers specified in the tenants application server or drives section. These can be a different servers per tenant. So there is no need for having like a global DNS service spanning all individual tenants.

The global DNS servers are used if there is no tenant specific DNS configuration or for the resolution of names linked to the appliance (for example proxy server, awingu update server, ...)

<table>
<thead>
<tr>
<th>Communication</th>
<th>From</th>
<th>To</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installer: TCP port 8080</td>
<td>Install PC</td>
<td>Awingu</td>
<td>Only needed once during the first time installer. Firewall rule can be removed after the installer has finished**</td>
</tr>
</tbody>
</table>
| NTP: UDP port 123  | Awingu   | NTP servers | Mandatory. Awingu needs to connect to a time server to make sure clocks get synchroized. In most cases the AD also can function as NTP server. If no internal NTP server is available also a public one like pool.ntp.org can be used.
| DNS: UDP port 53   | Awingu   | All DNS servers. | Mandatory. Awingu needs to do name resolution of all the DNS servers specified in the configuration. So both the global DNS servers as the optional DNS servers specified in the tenants |

When using an external database and if this database is not in the DMZ network also make sure all nodes have access to the database!
Install Awingu appliance

The Awingu appliance is available for on-premise installations or cloud installations. In this getting started guide we guide you through the installation of a single node Awingu. Please note that Awingu can also scale out vertically by adding more nodes to the Awingu cluster. Details on multi-node deployments are described in the admin manual.

There is no difference in features and functionality between running Awingu on-premise or in the cloud. Only the way to get and install the image is different and will be shortly discussed in this chapter. Once the image is deployed all the configuration is exactly the same.

Awingu Sizing Requirements:

| Minimum sizing is 2 CPU, 4 GB RAM & 80 GB Harddisk |

In a single node set-up, all processes are running on a single VM (frontend role, backend role and database role). This architecture can support only a limited number of the concurrent users. This has resulted in the following deployment recommendations:

- **Light Concurrent User:** User that has 1 RDP stream open and does not use the file operations heavily. This is typically the case when publishing VDI's or when all remote apps in a collection are **merged into a single RDP stream**.
- **Heavy Concurrent User:** User that has 3 RDP streams open, 10 accesses to reverse proxied web applications and does a number of file operations per hour per user.

<table>
<thead>
<tr>
<th>Concurrent Light Users</th>
<th>Concurrent Heavy Users</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 vCPU + 4 GB memory</td>
<td>100</td>
</tr>
<tr>
<td>8 vCPU + 8 GB memory</td>
<td>500</td>
</tr>
</tbody>
</table>

*3.7 GB like on some public clouds proposed is not sufficient

If you need more than 100 concurrent users or 300 concurrent applications install multiple appliances and run them together in a multi instance setup. See Awingu Administration guide for more information.

For deployment on private clouds / on-premise infrastructure:

The Awingu image can be downloaded from https://download.awingu.com/

From here the appliance images for most hypervisor technologies can be downloaded. This includes:

- VMWare / ESXi (5.5 and higher) both a vmdk and ova file are available
- Hyper-V (2012 and higher) a vhd file is available
- KVM / Openstack a QCOW2 file is available

Please download always the 'latest' version if you start a fresh install, the older versions are made available for users that would like to extend their existing Awingu cluster with additional nodes.

Our admin guide contains details on how to import the different images on the different hypervisors. Important to know is that once the image has been booted, it will broadcast a DHCP request and will honor the IP information it will receive. You will also be able to access the console to change the IP address to a static address. After the IP has been correctly set, you will be able to go through a web-based installation wizard to perform the installation.
For deployments on public clouds:

Azure Cloud:
On Azure the image can be deployed from the Azure Marketplace. You can simply add a New resource and search for Awingu to find the appliance in the marketplace. When you search for Awingu you will find 2 solutions in the Azure catalog:

- **Awingu**: Is the standard Awingu appliance, similar like the images you can download for on-premise. Installation still needs to be done and you need an existing windows back-end to connect with.
- **Awingu All-In-One**: Is a template that not only installs an Awingu image but it also setups for you a new AD server and at least one terminal server. Also all configuration and installation of the Awingu and Windows backend is done automatically for you so that at the end of the installation you have a full working green-field setup available.

Amazon Web Service (AWS):
Images for Amazon Web Service subscription can be readily deployed via the links provided on https://download.awingu.com

Google Compute:

Navigate to https://repo-pub.awingu.com/appliances/latest/gce/ in your web browser and download the most recent .tar.gz file.

You can import this image file into your Google Compute environment by following Google's official instructions. https://cloud.google.com/compute/docs/import/import-existing-image

After importing the image, create a new VM instance using this image, you will then be able to connect to the appliance's IP address (followed by port 8080) in your browser to configure the Awingu appliance.
Prepare your windows back-end

We assume in this getting started manual that there is already an existing windows back-end available. If this is not the case please do this first or use the Awingu All-In-One solution on Azure to deploy & setup everything for you.

Active Directory

Awingu doesn’t have any build-in user database and falls back to AD or an external IDP for this. Important to mention is that if users need/can change their passwords through Awingu the AD server needs to be equipped with an SSL certificate and configured to listen on SSL. In this getting started guide we will describe how a standard login against AD needs to be setup. For more complex scenario's like Single Sign-On with an external IDP like Azure, ADFS, Google, ... please have a look at the admin guide in the section on setting up SSO.

Awingu doesn’t require any specific functional AD level although to have features working like auto-import from AD we recommend a minimum of 2008R2. If the version is lower it might be needed to do manual imports vs automated imports via AD from certain resources like users, groups or application servers. More details on this are in the Awingu Administration guide.

Awingu is not compatible with Azure AD Only. Incase of using Azure AD for Single Sign-On (SSO) a real AD is still needed to join the application servers into the domain and to fetch the security groups.

In case there is no local AD the only solution is to install the Azure AD Directory Services feature (AADDS) and link the application servers and Awingu to these Azure AD DS servers.

Also to enable the auto-import of resources we need a binduser and password. This can be a normal, non-admin, account that doesn't need any privileges except that the password should not expire. No write access to the Active Directories is necessary. During the authentication in Awingu the user & password credentials provided will be used to do an LDAP or LDAPs bind with the AD server. If successful Awingu will get a list of all groups the user is member of.

Also it is recommended to put all users that need to become Awingu admin (= have privileges to make changes to the awingu configuration) into a dedicated security group.

Group Policies

There are a few group policies which are required for a correct end-user experience.

These are the following:

- Computer Configuration / Policies / Administrative Templates / Windows Components / Remote Desktop Services / Remote Desktop Session Host / Connections
  - Restrict Remote Desktop Services users to a single Remote Desktop Services sessions: Disable. (see remark below)
- Computer Configuration / Policies / Administrative Templates / Windows Components / Remote Desktop Sessions Host/Session Time Limits:
  - Set time limit for disconnected sessions: End a disconnected session in 1 minutes
  - Set time limit for log off of RemoteApp sessions: RemoteApp session log off delay Immediately
- Computer -> Policies -> Administrative Templates -> Windows Components -> Remote Desktop Services -> Remote Desktop Session Host -> Connections
  - Allow remote start of unlisted programs: enabled

Via the "session merge" option in the app publication, different apps can be merged into existing remote app sessions avoiding that for each app a new RDP stream needs to be started resulting in loading GPO's and profiles each time.
Next to the required GPOs for a correct functional working, there are also a couple of the GPOs that we have gathered within the Awingu Administration guide to restrict the security settings within a remote desktop session.

When using User Profile Disks or FSLogics Profiles make sure all apps from the same collection are session merged in Awingu

**Microsoft Application Server (RDS)**

When windows application should be delivered, we require the setup of remote desktop session host (RDSH) on the Microsoft windows backend. Awingu can integrated with windows server 2008 R2, Windows 2012, Windows 2012 R2 and Windows 2016. Windows application on the backend can both be published via RemoteApp or by starting an application within an RDP session. Delivering application via RemoteApp is the recommended approach. For older versions of Windows Server like server 2003, server 2008 only connections via RDP will be working, not RemoteApp.

Application that needs to be delivered should be installed on the application servers and operate within a multi-user context. When using the RemoteApp protocols the application can be directly published (as RemoteApp) from within the Server Manager. Please refer to the Awingu Administration manual for detailed description on how to publish remoteApps.

Awingu supports both connections directly to a session host as also making connections to an RDS broker.

**VDI / Full desktops**

Next to applications Awingu can also connect via RDP to full desktops. When using Windows XP, 7, 8, 10 or servers without RDS session host roles installed the logins will be limited to 1 user per server/client.

Awingu is also compatible with Windows 10 Multi User (aka WVD). in that case multiple users can connect to the same Windows 10 image.

Also important to notice that prior to access via RDP a windows client OS the users have to be added explicitly to the allowed remote access users. By default only domain admins are allowed to do RDP login onto a windows client OS.

**Files Service Integration**

The Awingu appliance allows to integrated towards existing files servers. For windows filesshares it is important that those filesshares are part of the same domain as to which awingu is configured. All access to the files will be done with the identity of the user itself.
The Awingu Installer

Once the Awingu appliance is been deployed on premise or in the cloud the first thing that needs to happen is to run once the Awingu installer. During the installation the Awingu appliance will initiate and setup all services.

Use a web browser to go to http://<IP Address of your appliance>:8080/ to begin the installer. The installer itself is a 5 step wizard:

![Eula](image)

### Eula

If deployment was successful you should see the EULA as a first step:

Please read the EULA and then check the “Yes, I have read and hereby accept the above license terms and conditions” to proceed.

### Management User

Next step is to setup a Awingu Management User.

![Management User](image)

Important to know is that this user can’t be changed and should not exist in your AD domain. Password can be changed later on.

This Management User will be able to login at any time and alter configuration settings. This management user will not be able to launch streamed applications or access drives. This user is not taken into account for licensing and does not require a one-time-password (OTP) to sign-in. It is advised not to use this Management User, other than for the initial install or in case of emergency.

The user automatically logs out after 15 minutes!

### DNS & NTP

At step 3 of the installer we need the IP address(es) of the DNS and NTP servers. In general we recommend to use the AD servers as DNS servers as such they can also resolve the internal names of the application servers used.

Host name can not be changed afterwards. When planning a multi node deployment best set directly the host name to a meaningful name.

### Databases

Step 4 will give you the option to use the build-in database or use an external database. External databases are needed when you would like to do a multi-node deployment of Awingu. Multi-node deployments are needed for +100 concurrent user setups or for setups that require high availability. Important to mention is that you can’t switch afterwards from an internal to an external database without reinstalling everything. Details on the connection strings and supported flavors for an external database can be found in the admin manual. For this getting started manual we assume a single node Awingu deployment with build-in database.
Summary

Step 5 is the summary of the configuration and after clicking on "finish" the installation will start. Please take into account that, depending on the type of storage used and the overall performance of the system this installation can take between 10 to 30 minutes.

If installation is successful the installer will disappear and a login prompt will be presented. The install service on port 8080 will also no longer be accessible.
Connect Awingu to your Active Directory

First login & System Settings

Once the installer has finished you will be able to login with the build-in username and password into the Awingu appliance.

To do so go to the IP or DNS name of the awingu appliance on port 80. (example http://172.16.0.10/), the domain field must be left blank. The first time you login you will need to accept the privacy statement.

Notice the port 8080 where the installer was running, will not be available anymore.

After logging in you will see the Awingu Workspace with 3 available applications:

1. System settings: Here you can manage your environment and make the necessary changes to the configuration
2. API DOCS: Awingu is fully API driven, this will open the API interface and show you the API documentation
3. Dashboard: Over here you can see all audit & monitoring information

Open the system setting by clicking on the icon.

Add a Domain

Awingu has no own user database and needs to be connected to an existing AD or LDAP server. In this getting-started guide we will go through all steps needed to link the Awingu appliance to an existing AD server.

Start by opening the Domains page (System Settings  Global  Domains)

Click on Add and scroll-down to add the first tenant/domain:

Following parameters are mandatory:

- NetBIOS Domain Name: NETBIOS domain name (e.g. COMPANY)
- Name: This is the internal name used in Awingu for this domain. Default is the same as the NetBIOS.
- FQDN for UPN: The FQDN for the domain itself, not for the AD (e.g company.local).
- DC/LDAP server: FQDN or IP address of the Domain Controller or LDAP Server. E.g. ad01.company.local. Multiple servers can be entered comma separated. The first server will always be tried as first one during login.
- Base DN: When a user signs in, this base distinguished name (DN) is used to bind via LDAP to the Domain Controller/LDAP server. This can be used to filter access based on organizational unit (OU).
  - Example without OU restriction: dc=company,dc=local
  - Example with OU restriction: ou=Employees,dc=company,dc=local
- LDAP over SSL?: Set to yes if there is an SSL certificate on Domain Controller or LDAP Server. This is not the case by default. Also note
that LDAP over SSL must be enabled to be able to make changes to passwords via Awingu

In case of doubt of the correct values you can validate them via following power-shell command on the AD server:

```
PS C:\temp> Get-ADDomain | select NetbiosName, DNSRoot, InfrastructureMaster, Distinguishedname | Format-List
NetbiosName : COMPANY               <--- NetBios Name
DNSRoot     : company.local        <--- FQDN for UPN
InfrastructureMaster : ad1.company.local  <--- DC/LDAP server
Distinguishedname : DC=company,DC=local <--- base DN
```

Optional parameters:

In the next step we will add application servers to awingu and also import users and security groups. To do this automatically from AD Awingu needs a bind user & password. If not set auto-import will not work and these servers, users & groups will need to be created manually in Awingu. The bind user/password can be a normal user and doesn't have to be a domain admin user.

- **Bind Name**: The username of the service account. (example: serviceuser). No domain or FQDN need to be added.
- **Bind Password**: The password required to authenticate the service account.

The "create bind name", "find groups" & "Privacy Policy Acceptance" parameter should not be changed. See manual for more information.

Multi-tenancy parameters:

Awingu supports multi-tenancy, which means that on a single Awingu (cluster) you can run multiple Awingu instances. To do so add multiple domains. To support this multi-tenancy some extra parameters have been added. In this getting started guide we only add a single tenant but if later extra tenants are added following parameters will be needed:

- **Host Headers**:
  - Single tenant: leave blank
  - Multi tenant: List all public hostheaders/dns names (remote.company.com) linked to the domain/tenant, when going to one of these hostheaders Awingu will automatically load the correct tenant

- **Administrative Domain**:
  - Single tenant: leave on Yes
  - Multi tenant: if set to yes all admins in this tenant will also be able to manage the other tenants. So only do this for the tenants where relevant. You need at least 1 tenant with administrative domain set to yes.

- **DNS Server**:
  - Single tenant: Leave blank, in this case the overall DNS server will be used.
  - Multi tenant: When leaving blank the overall DNS server will be used, so it must be able to resolve the server names in this domain. If this is not the case, specify one or more DNS servers that contain the DNS information of this tenant.

Test the login

When finished click on the "add" button at the bottom of the page. Awingu will now add the tenant and make the link with the domain controller.

Test if the settings are done correct by logging in with a normal domain user:

1. close the "system settings" tab in the browser
2. in the awingu workspace click at the bottom on the build-in admin user and select log out
3. try to login with a normal domain user. You can use either the sAMAccount (domain\user or user) or the UserPrincipleName (user@company.local)

If it works you know your domain settings are correct!

Labels

All configuration in Awingu is done via "labels". Labels is a combination of a key and a value.

There are different kinds of labels:
User Labels: these start with username or group and can be used to link to a specific user or group in AD. For example the label group:Administrators refers to the AD group Administrators, the label username:steven refers to the user steven in AD.

Server Labels: these have a more free format but for example every time you import or add an application server the appserver:ServerName is created so it can be directly used in the configuration.

Next to these labels also some build-in labels exist like "all:“, "admin:“, "record:“, ....

Most easy way to add user or group labels is to import them automatically. To do this goto the Manage Labels and click on the import from AD buttons. Once the groups and / or users are imported Awingu will keep tracking them so when an user is added to a group in AD no extra actions are needed on Awingu side.

More information on the labels in the admin Manual.

If "import from AD" doesn't work (there is a red error message) this is probably caused by a wrong user / password in the domain configuration. The import from AD uses the bind user specified in the domain settings.

Finetune the access rights

Login with the build-in admin and open the "system settings".

Next step is to further finetune the access. Currently all domain users can login to Awingu and only the build-in user can manage the configuration.

Assume you have 2 security groups in AD to which you would like to limit access:

- "Awingu_Users" contains all users that are allowed to login
- "Awingu_Admins" contains the users that can be admin on Awingu

First start by importing these groups labels (top menu Manage Labels).

Once you have the labels imported goto the user connector (top menu Configure User Connector)

Over there you will find the login permissions. Add the group or user labels to the "domain administrators" that need to be admin on Awingu. In this example we have added the group "Awingu_Admins". By default the "sign in White List" is set to "all". This means that all users in the domain can login. In this example we have restricted the access to the Awingu environment to the users that are part of the "Awingu_Users" and "Awingu_Admin" groups.

Login Permissions

Domain Administrators: [group:Awingu_Admins]

Sign in White List: [group:Awingu_Admins] [group:Awingu_Users]

User Profile Defaults

Keyboard layout: English United States
Once this is done logout with the build-in user and login with a user that is part of the "Awingu_Admins" security group. As you will see this user has now also access to the Awingu admin configuration and there is no need anymore to use the build-in user for normal day-to-day configuration.
Enabling SSL

As Awingu is 100% web based it is important to encrypt the traffic via SSL. To do so their are different possibilities. One of them is to use the internal SSL-Offloader of Awingu which we will explain in this chapter.

If you plan to use an external SSL-Offloaded check the full manual on how to configure it correct so the websockets used in Awingu are passed correct. Pay attention to the needed headers that need to be passed.

To enable SSL offloading you will need a certificate to encrypt the web traffic.

There are 3 ways to get a certificate:

1. You buy / have already a valid certificate issued by a certificate authority. In that case make sure it is in CRT/Key format. If not use openssl to change it to the correct format. Both individual as wildcard certificates are supported in Awingu. Check FAQ in support for help with converting certificates if needed (for example from pfx to crt/key)
2. You generate a self signed certificate. This is not recommended as some web browsers (especially apple devices) not always accept them
3. You use the build-in integration of Awingu with Letsencrypt. Letsencrypt is a free certificate authority and Awingu has done a full integration with it so it fetches and updates automatically certificates

Before uploading the bought/self-signed certificate or generating the letsencrypt certificate also make sure the public DNS name you would like to use is correctly configured.

To enable the SSL first open the certificate page (System Settings Global Certificates)

Click on Add at the bottom to add a certificate. Know that Awingu can handle as many certificates as needed so you are not limited to a single certificate.

In case of an own certificate upload the CRT & Key file. Awingu will extract automatically the certificates name:

The Awingu appliance needs to be accessible from the outside world (internet) via port 443 and in case of letsencrypt also from port 80! Port 80 needs to remain open for certificate renewals on letsencrypt. If it is not open from remote the certificate renewal won't work and the appliance will stop working after 3 months!

Awingu has a build-in redirect feature that will automatically redirect incoming http traffic to https. So no end-users will be allowed to connect via http to the appliance.

In case of letsencrypt (automatic) specify the public DNS name that can be used to access the Awingu appliance:
Once done click on "Add". The Awingu appliance will now install the certificate and be both accessible on http as on https.

Redirect http to https traffic

The last thing to do is to enable the redirect feature that all incoming traffic to http (port 80) is redirected to https (443)

To enable this redirect you must be connected over https. So first logout of the Awingu environment and relogin over https.

Open the connectivity page (System Settings Global Connectivity) and change the SSL Offloader from "Optional HTTPS" to "Internal SSL offloading with enforced HTTPS":

SSL Offloader

State

- Optional HTTPS
  You can access Awingu through both HTTP and HTTPS, but the specific security measurements for HTTPS (e.g. Cookie Secure Flag) won't be applied. This setting can be used with both the internal and an external SSL offloader.

- Internal SSL offloading with enforced HTTPS
  All HTTP traffic to Awingu will be redirected to HTTPS, which will be handled by the internal SSL offloader of Awingu. This enables specific security measurements for HTTPS (e.g. Cookie Secure Flag).

- External SSL offloading with enforced HTTPS
  Awingu accepts non-SSL traffic, but specific security measurements for HTTPS (e.g. Cookie Secure Flag) will be enforced. You need an external SSL offloader to access Awingu.

Cancel Apply
Multi Factor Authentication

Awingu can be configured to do a second factor authentication, next to the login/password check.

There are 3 possibilities to activate MFA:

1. Use one of the Awingu built-in solution:
   a. Awingu OTP: Counter based
   b. Awingu OTP: Time based
2. Connect via a build in connector to:
   a. Azure MFA (requires the Azure MFA server to be installed)
   b. DUO security
   c. SMS passcode
3. Use the radius connector to connect to any MFA solution that supports radius (symantec VIP, RSA, Vasco, ...)

For option 2 & 3 check the admin manual for further configuration. In this getting started guide we will configure the build-in solution.

The counter based tokens are not supported by Microsoft Authenticator. If you would like to use Microsoft Authenticator then use the Time Based OTP solution.

Google Authenticator supports both the Counter & Time Based OTP.

To enable the MFA feature open the user connector page (System Settings Configure User Connector)

Scroll down to the "Multi Factor Authentication" section and set the mode to "Awingu OTP: Time based"

Optional you can:

1. Set a list of ip addresses or ranges of ip addresses for which the MFA will not be asked. This could for example be used to exclude MFA validation from in the office
2. Set a list of user or group labels that are excluded from using the MFA
3. Enable the Trusted Browser feature. If done the user will have the option to disable for 30 days the MFA check on a specific device / browser

To check if it works login again with any user from an ip address that is not in the white list. The first time you connect you should get a QR code on the screen:
Open your Authenticator app of choice, scan the QR code and enter the 6 digit key. As of now your smartphone is linked to your account.

In case somebody needs to reset the token because go back to the "multi-factor authentication" part of the "user connector" page and click on "manage user token count":

Click on the "reset" button for the user of who you would like to reset the token. Once done the next time the user connects he will get the QR code back and could link it to a new device.
Publish apps and desktops

**Adding Application Servers**

Before we can add applications to Awingu we first need to add the application servers (System Settings  Manage  Application Servers)

We assume you have configured a bind user / password in the "Connect Awingu to your Active Directory" so we can use the "Import from AD" option.

Select the servers you would like that Awingu can use to connect to. Once the servers are selected in the “Set Default Settings” part specify the number of concurrent connects that can be made to this machine and set the state to enabled. When finished click on Import. As of now Awingu knows these servers and they can be used later when creating applications or desktops in Awingu.

When adding Windows Client OS (like Windows 10 or 7) or when adding servers that do not have terminal server installed on them the maximum connections needs to be set to 1

**Adding an application/VDI**

Before you start adding applications or desktops make sure the application servers are imported and if you like to restrict access to a certain group or list of users make sure the user labels have also be imported!

To add any type of application (remote app, rdp app, web app, ...) or VDI, open the Applications page (System Settings  Manage  Applications) and then click on "add"
Awingu currently supports 5 types of applications:

- **Desktop Applications**: Use this if you would like to make a full desktop connection to a Windows Server or Client OS.
- **Remote Applications**: Stream a Microsoft Remote Application. For this the app first needs to be added to an existing RDS collection.
- **Web Applications**: Link to a public available website.
- **Reverse Proxied Web Applications**: Link to an internal web application. Awingu will act as proxy server for these kind of websites to make them also available from external.

For this getting started guide we will limit the documentation to a standard setup of the first 2 kinds of applications (Desktop & Remote). For the other 3 (RDP, Web & Reverse Proxied Web) or advanced settings please have a look at the admin guide.

**Generic Settings:**

Following settings are common for all types of applications:

- **Name**: The application name as it will appear in the Awingu user interface.
- **Description**: Description of the application, not visible to end-users.
- **Icon**: The application icon that will be visible to the end-user in the Awingu user interface. Only ICO, JPG and PNG are allowed. Maximum size is 100kb.
- **Categories**: This application will be shown in the selected categories.

**Desktop Application Settings**

When adding a desktop application the only thing left to do is to link the application to the correct user & server labels.

- **User Labels**: contains the list of all users and groups that are allowed to start an application. If all users can start it use the build-in "all:" label, if only awingu admins can use the app set it to "admin:". If you want to restrict it to a list of specific AD groups or users set it to the group and user labels that have been added before.

- **Server Labels**: contains the list of application servers on where the application can run. For desktops and single apps fill in at least one application server. When multiple application servers are entered, Awingu will do automatic round-robbing loadbalancing over the different servers. For web applications the server labels should be left blank.

So for example if you would like to make a specific desktop available for a specific user the, you first import the desktop via the "manage application servers" and import the user via "manage labels". This will create the correct labels that can now be assigned to the VDI.

**Remote Application Settings**

Before we can add the user & server labels for a "remote application" we need to specify which remote application needs to be started and if we need to link the application to specific filetypes.

- **Alias**: This corresponds to the alias (second column) of the published application in the collection.
• **File Types:** Linking apps to "file types" will allow the files to open directly from in the Awingu drives.


• **User & Server labels:** Just like for a full desktop link the "Remote Application" via the user & server labels to the correct users and servers.

Awingu can also connect to an RDS broker and use the broker functionality to do the loadbalancing. in this case don't specify the different session hosts in the server labels but set the value to the correct collection. See Awingu manual for more details (add the rdscollection label, link it to the broker and then link the rdscollection label to the app)

**Advanced Settings:**

Following settings can be used to further finetune the configuration. For more details have a look at the admin manual.

• **Unicode:** In general we recommend to leave the unicode to enabled for apps. If this is done then the keyboard is detected automatically. Only set unicode to disabled when you encounter strange behavior in typing in apps. When disabled user have to select in their profile the correct keyboard to work.

• **Labels** can be used to group specific apps for reporting

• **Auto start labels** can be used to select a group of users for who this application will automatically start when they login. So no need anymore to click on the icon, this application will directly start. Set the **start in foreground** option to enabled if you want that this app is also active

• **Concurrent Usage** can be set to disabled if you want that this application can only be started once per user. For example outlook should be restricted to a single start where word can be opened multile times by the same user.

• **Ask for Credentials** can be enabled if no single-sign-on is needed and the user has to specify the credentials for this application or desktop. Can be used to execute applications as an other identity then the one you have used to login

• **Notifications** can be disabled if you don't want to receive any notification from this application.

• **Session Merge** allows you to merge this application into an existing RDP session. If there is already an remote app running on a server that is also in the server label list of this application and the feature is enabled, then Awingu will not start a second RDP session but merge this application into the existing RDP stream of the first application.

• **Minimum Size** allows you to specify a minimum size for this application. When the screen size is smaller then the minimum size of the application Awingu will render the application bigger then the screen size and allow you to move the application window in the screen window.
Publish drives

Awingu drives allow you to access from in Awingu directly network drives. By doing so we can for example open files directly in one of the published apps. The Awingu drives can also be used to upload/download files or to share them.

To add a (network) drive go to the drives page (System Settings Manage Drives) and click on the add button at the bottom of the page.

Adding Drives is similar like adding applications. If you would like to restrict access to a specific group of users make sure the necessary user labels are created upfront.

The name should be set to the display name in Awingu.

The backend specifies which kind of storage to integrate with. For Onedrive or Webdav see the admin manual on how to configure them.

In this getting started guide we will assume the storage back-end is CIFS (SMB).

The URL parameter is used to browse from the Awingu appliance the storage. Make sure it is in the format smb://<ip_or_fqdn>/path/  (Example: smb://file-server.domain.local/share )

The UNC parameter is used to open the file from the application server. Make sure it is in the format \<ip_or_fqdn>\path\ ( Example: \\fileserver.domain.local\share )

The <username> tag can be used in the URL & UNC path to go directly to a user specific directory. For example \\
fileserver.domain.local\users\<username>\documents will open directly every user his own documents folder.

The last step before the drive will be available in Awingu is to link it to the correct user labels:

User Labels contains the list of all users and groups that are, just like with the applications, allowed to access this network drive. If all users can access it use the build-in "all:" label, if only Awingu admins can use the app set it to "admin:" . If you want to restrict it to a list of specific AD groups or users set it to the group and user labels that have been added before.

Important to mention is that the access to the storage is determined by the access rights on the backend. So even when in Awingu "all:" users can access the drive it will only work if those users also have the necessary permissions on the storage backend itself.
Customize the branding

Awingu can be easily customize to the desired branding. To do so open the branding page via System Settings  Configure  Branding.

In the **general section** you can select the colors and also if you would like to have a "plain background" color or the one with the "polygon style" fading color background.

For the **logo sections** if you would like to use a custom logo you not only need to upload a new image but also set the "active wide logo" and "active square logo" option to "custom"

For the **login page section** if you would like to use a custom background image first also set the "active background" to custom. There are 2 images that can be uploaded. One for large screens and one for smaller screens. Finally also the login text can be customized. The text allows html input.