Application Note

Active Directory Federation Services deployment guide

Document version: v1.1

Last update: 20th January 2014

RULE YOUR NETWORK!®

EXCELIANCE

Purpose

ALOHA Load-Balancer deployment guide for Microsoft **ADFS** and **ADFS proxy** services.

Complexity

ALOHA Versions concerned

- Aloha 4.2 and above

Changelog

- 2014-01-20: improved health checks
- 2013-11-05: Initial Version

Introduction

About Exceliance

Exceliance is a software company, editing the **Application Delivery Controller** named **ALOHA LoadBalancer**.
Headquartered in Jouy-en-Josas (Yvelines), **Exceliance** teams (including R&D and technical support) are based in France. **Exceliance** has currently around 100 customers in the banking, retail groups, energy and e-commerce industries and the public sector. **Exceliance** solutions are also used by many hosting providers. The **ALOHA Load-balancer** is designed to improve performance, guarantee quality of service and ensure the availability of critical business applications, by dynamically balancing flows and queries on the company's various servers.

About ALOHA Application Delivery Controller

The **ALOHA Load-Balancer** is designed to load-balance at layer 4 and control application delivery at layer 7. It is designed to integrate simply and quickly into any environment or architecture.
The main **ALOHA** benefits:

1. Guarantee high availability of your applications and services as well as improve web application performance
2. Makes infrastructures scalable and reliable
3. Simplifies architecture: IPv6 frontend, SSL offloading, layer 7 routing

Developed using **HAProxy** open source load balancing software, **Exceliance** solutions are known for their processing performance, reliability and wealth of features. Offered at more affordable price than other commercial solutions, they are easy to deploy and administer.

ALOHA Load-balancer is available either as a physical appliance or as a Virtual Appliance. The Virtual appliance can run on top of the most popular hypervisors available on the market.

About this guide

This guide first explains in the main lines how **Microsoft ADFS services** is designed and what benefits an ADC can bring.
The guide also provides with configuration templates to setup the **ALOHA Load-Balancer** for Microsoft **ADFS** and **ADFS proxy** for the two most common architectures.
The lastest version of this guide can be downloaded from Exceliance website: `http://www.exceliance.fr/en/`.

Appliance supported

All **ALOHA Load-Balancers** appliances, both physical and virtual, can be used with Microsoft **ADFS** and **ADFS proxy**.

Microsoft ADFS version supported

ALOHA load-balancer can be used with the following versions of Microsoft **ADFS protocol**:

- ADFS v2.1 (windows 2012)
- ADFS v2.0 (windows 2008r2)

Disclaimer

The **ADFS** configuration tips provided in this guide are purely informational. For more information about Microsoft **ADFS** tools and how to use them, please refer to Microsoft web site which is fully and properly documented.
This guide does not provide information on how to install and setup an **ADFS** services.

Introduction to Microsoft Active Directory Federation Service

ADFS is Microsoft solution to provide simplified, secured identity federation and Web **Single Sign-On** (SSO) capabilities for end users who want to access applications within an AD FS-secured enterprise, in federation partner organizations, or in the cloud.
It is a software component that can be installed on Microsoft Windows Server operating system.

It provides the following accesses:

- Company's employees or customers with a web-based SSO when accessing internal applications
- Company's employees or customers with a web-based SSO when accessing resources in any federation partner organization
- Company's employees or customers with a Web-based SSO when accessing internal applications from remote locations
- Company's employees or customers with a web-based SSO when accessing resources or services in the cloud

ADFS server roles

There are two types of server roles in an ADFS architecture:

1. ADFS server
2. ADFS proxy server

ADFS server role

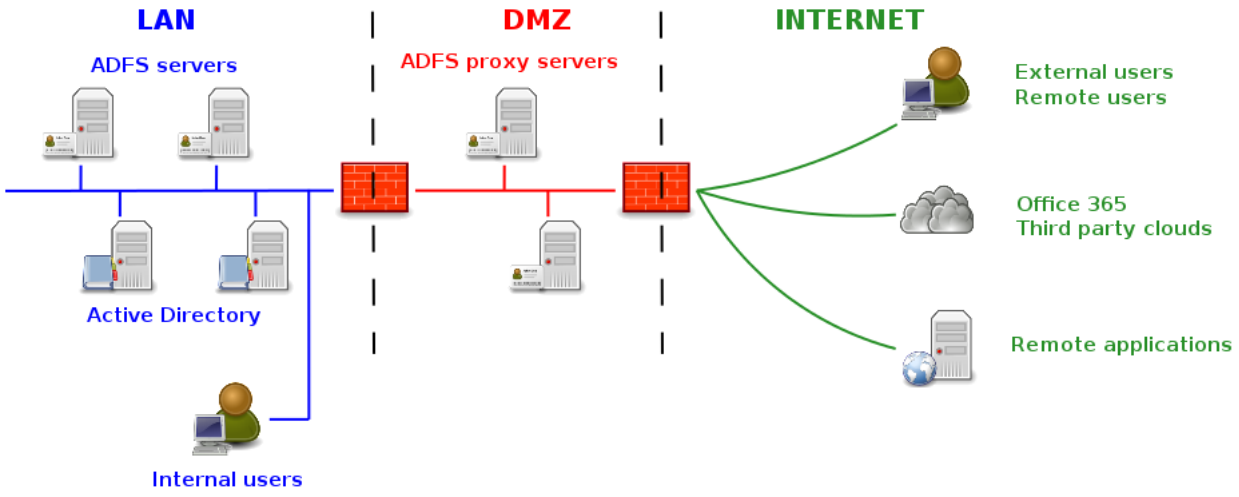
This is the **main** component and it is **mandatory**.
It is responsible for delivering user authentication. It must be part of the **domain** and able to connect to the **Domain Controller**. It authenticates users from multiple domains via Windows Trust.

ADFS proxy server role

It brokers a connection between external users and internal **ADFS servers**. It acts as a reverse proxy and resides in organization's perimeter network (aka DMZ) and does not need to be a member of the domain.
This role is **optional** but recommended when opening the **ADFS** infrastructure to users based outside organization's trusted network: it improve security.

ADFS infrastructure

The picture below shows where the different **ADFS** server roles are installed and how they interract together.



Flow for Internal users

When an internal user wants to access a claims aware application, the following flow happens:

1. The **internal user** accesses the claims aware **application**
2. The **application** redirects the user to the LAN **ADFS server**
3. The **ADFS server** authenticates the **internal user**
4. The **ADFS server** performs an HTTP post to the **application** where the user gains access

The redirects and posts are performed using a standard HTTP 302 Redirect and HTTP POST respectively.

Flow for External users

When an external user wants to access a claims aware application, the following flow happens:

1. The **external user** accesses the claims aware **application**
2. The **application** redirects the **external user** to the **ADFS proxy server**
3. The **ADFS proxy server** connects to the **LAN ADFS server** and forwards it the authentication request
4. The **LAN ADFS server** authenticates the user and forward the response to the **ADFS proxy server**
5. The **ADFS proxy server** performs an HTTP post to the **application** where the user gains access

ADFS ports and protocols

The table below summarizes the ports and protocol involved in an **ADFS** infrastructure:

Role	Port	Protocol
ADFS server	TCP/443	HTTPS
ADFS proxy	TCP/443	HTTPS

Server affinity

ADFS is a web service and does not require server affinity!

Why using a load-balancer with ADFS services

First of all, even if **ADFS** provides service arrays, to ensure high-availability, it does not provide any load balancing mechanism.
This means we need a third party appliance to balance traffic amongst **ADFS Servers** and **Proxies**.

A hardware load-balancer brings the following benefits to Microsoft **ADFS** infrastructure:

- **Fast Transparent failover**
  Nobody will notice a server has failed since the Load-Balancer is able to quickly and transparently redirect users to a healthy server.
- **Application aware health checking**
  A load-balancer provides application layer health check which provides the status of the service itself and are more efficient than a simple ping.
- **SSL bridging**
  A load-balancer can inspect ciphered traffic between a client and an **ADFS server** or **Proxy** to improve protection, reporting servers and URLs response time, etc...
- **Scale up**
  Building an architecture with a load-balancer allows scale up: adding new servers can be done easily without production outage.

Load-Balancing ADFS servers

The configuration template provided below below is the simplest way to load-balance **ADFS servers**.
For more complex scenarios, such as brute force protection, please contact **Exceliance team**.
This configuration is to be deployed on an **ALOHA Load-Balancer** deployed in the **LAN** zone.

```
frontend ft_adfs
  bind 10.0.0.90:443 name adfs
  mode tcp
  log global
  option tcplog
  maxconn 1000
  default_backend bk_adfs

backend bk_adfs
  balance roundrobin
  mode tcp
  log global
  option tcplog
  default-server inter 3s rise 2 fall 3
  option httpchk GET /adfs/ls/IdpInitiatedSignon.aspx HTTP/1.0\r\nHost:\ adfs.domain.com
  http-check expect string Sign-In\ Page
  server adfs1 10.0.0.101:443 maxconn 1000 weight 10 check check-ssl
  server adfs2 10.0.0.102:443 maxconn 1000 weight 10 check check-ssl
```

Don't forget to update the following information:

- **bind**'s IP address
- **servers** IP addresses
- **adfs host name** for health check

Load-Balancing ADFS proxy servers

The configuration template provided below below is the simplest way to load-balance **ADFS proxy servers**. For more complex scenarios, such as brute force protection, please contact **Exceliance team**. This configuration is to be deployed on an **ALOHA Load-Balancer** deployed in the **DMZ** zone.

```
frontend ft_adfs_proxy
  bind 192.168.100.90:443 name adfs_proxy
  mode tcp
  log global
  option tcplog
  maxconn 1000
  default_backend bk_adfs_proxy

backend bk_adfs_proxy
  balance roundrobin
  mode tcp
  log global
  option tcplog
  default-server inter 3s rise 2 fall 3
  option httpchk GET /adfs/ls/IdpInitiatedSignon.aspx HTTP/1.0\r\nHost:\ adfs.domain.com
  http-check expect string Sign-In\ Page
  server adfs_proxy1 192.168.100.101:443 maxconn 1000 weight 10 check check-ssl
  server adfs_proxy2 192.168.100.102:443 maxconn 1000 weight 10 check check-ssl
```

Don't forget to update the following information:

- **bind**'s IP address
- **servers** IP addresses