Azblink System Deployment Guideline for Capacity Planning

Abstract

This document is to describe how to cope with performance issues by expanding the capacity of Azblink Voice/Video/Messaging system. The analysis will be categorized into the following areas: application (voice, video, and text messaging), user authentication, storage space, and networking.

Table of Contents

Illustration Index

Introduction

Azblink Unified Communications System bundles voice, video, and text messaging along with VPN into a single host. It is sitting on the boundary between corporate private network and Internet. The voice, video, and text messaging are running over VPN from the APP on the mobile devices to connect to the server. With the tunnel established by the VPN, other applications could take advantage of this connection to exchange data with the other hosts in the office. The host itself can also connect to IMS (IP Multimedia System) of the telecommunication service providers and PBX (Private Branch Exchange) via SIP trunk to place/accept voice calls to/from other places.

However, a single host is with its own limitations no matter how powerful the hardware it is equipped with. For example, the TCP/UDP ports binding on an IP address can be no more than 65535. This implies that you can not have so many concurrent connections on a host with single IP address. Thus, to some extent, it is necessary to have multiple hosts working together.

Users associated with the same host can place voice/video calls or send text messages to each other. But how to place voice/video calls or send text messages to the users on the other hosts? For voice and video, SIP trunk can be established between hots. For text messaging, those hosts can be joined together to work as a cluster or let them work under theirs own domains and allow server-to-server communication in XMPP. But different business requirements would drive to different deployment scenarios: for example, shall we allow file sharing via XMPP messaging? Or shall chat records be kept for audit purpose? We will discuss them in the following sections.

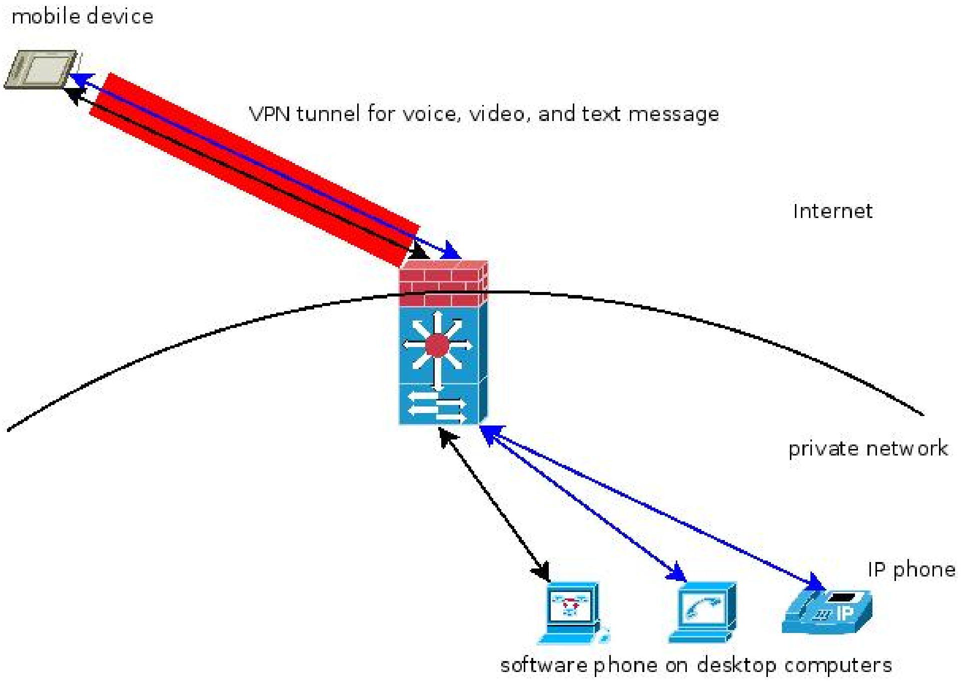
mobile device
VPN tunnel for voice, video, and text message
Internet
private network
IP phone
software phone on desktop computers

Illustration 1: Single Host for Application

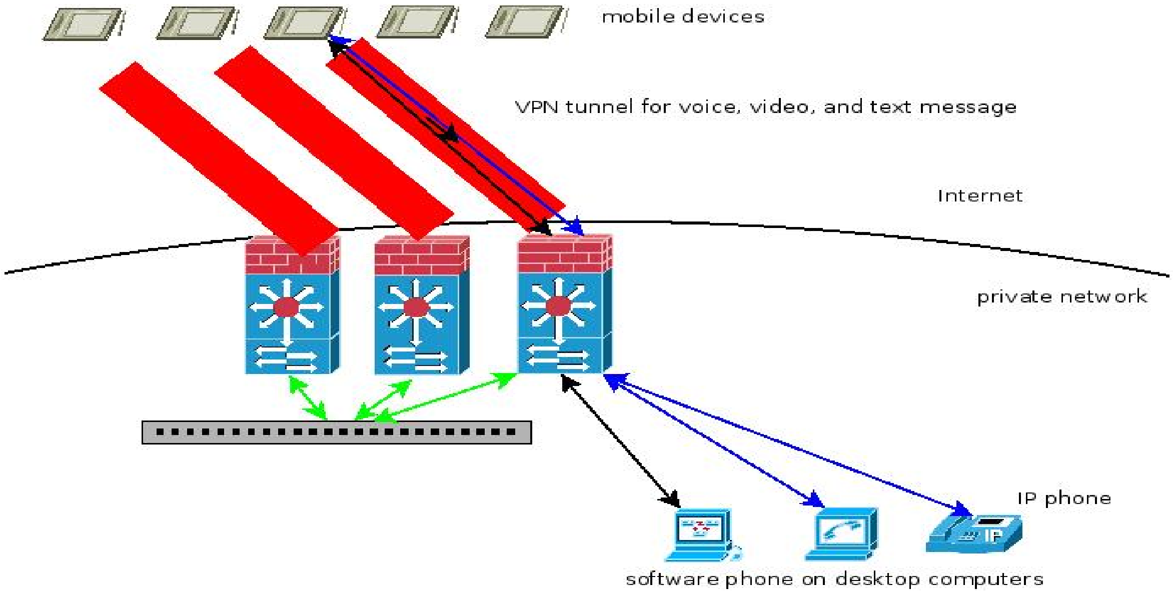
mobile devices
VPN tunnel for voice, video, and text message
Internet
private network
IP phone
software phone on desktop computers

Illustration 2: Multiple Hosts for Application

Basic Functional Blocks

As mentioned earlier, if there exist performance issues in single host scenario, we set up multiple hosts to alleviate performance bottleneck. However, simply stacking machines would not make those multiple hosts to function as one seamlessly. If you are familiar with Azblink Voice/Video/Text Message System, you should know that our SIP accounts for Voice/Video calls are with numerical numbers, and XMPP for text message system uses alphanumeric characters plus domain name as JID. And we associate the numerical number for SIP and JID in XMPP along with VPN key to a user. The user employs this VPN key to establish encrypted tunnel with the server, place voice or video calls by using numerical number as SIP account, and sending text message with JID as identifier in XMPP.

There exist the following action items while integrate multiple hosts together:

1. the dialing rules to the other host for voice/video calls associate with SIP trunks
2. joining all the hosts as a cluster for XMPP

On each node of XMPP cluster, it needs to have the knowledge of all JIDs in the cluster. It is not like SIP. In our implementation for SIP, each node does not have the knowledge of the user accounts on the other nodes; it is counting on the dial rules to route the calls to the other hosts.

And within XMPP, it allows file transfer from one user to another via http upload and download. In the scenario of single host, the uploaded file will be deposited in a directory (namely “/home/flv”) of the file system on that host, and the other user fetches the file from that directory. For file sharing in XMPP to work properly in multiple-host scenario, it is necessary to open the directory for the other hosts to access or result to external storage so that all the hosts in the cluster can access. Opening the directory on each host for the other hosts to access is with the following issues:

1. the private network behind the hosts shall be open to VPN (because XMPP users will fetch the uploaded files from VPN)
2. the VPN on each host should use different private network address

3. the routing entries to each VPN subnets shall be added into the IP routing table on each host

Thus, it would be easier to use external storage and mount the space provided by that external storage to the upload depository directory ("/home/flv") of each host so that the function can work as in the case of the single host scenario.

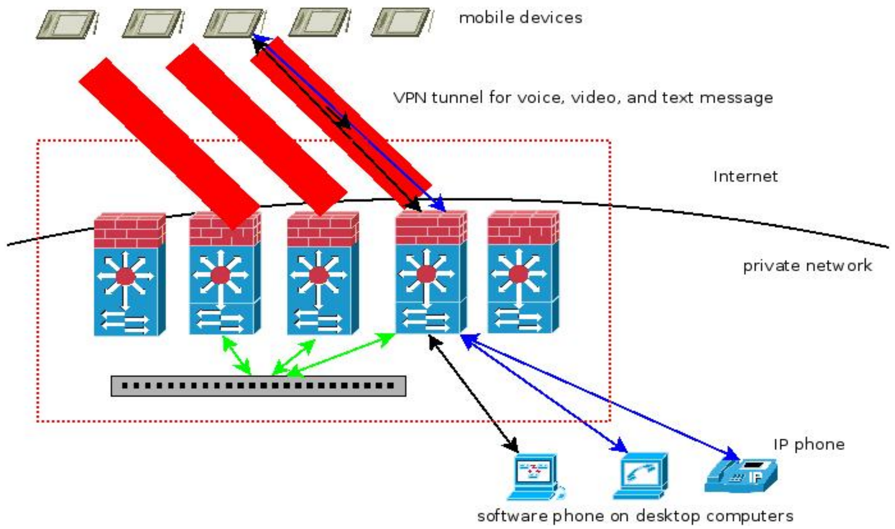
mobile devices
VPN tunnel for voice, video, and text message
Internet
private network
IP phone
software phone on desktop computers

Illustration 3: Simply Stacking the Machines

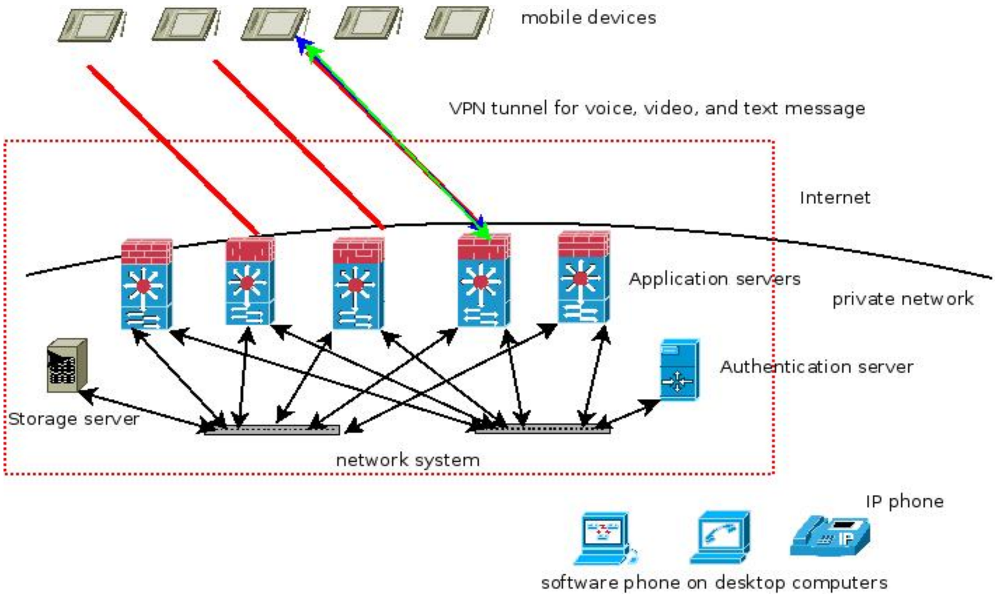

Illustration 4: Using External Authentication Server and Storage Server

From the requirements mentioned above, it would be easier to have external authentication server with all the account information and external storage server for all the application servers. Azblink Voice/Video/Text Message system provides authentication via LDAP server and NFS or Samba to access external storage.

If all the text messaging and voice calls shall be recorded for audit purpose, another storage system is needed. However, the operation principle is similar to what we have discussed above; in other words, mount remote file system via NFS or Samba to a specific local directory ("/home/chat") . For the simplicity of the discussion, we just use "upload/download" as an example for the performance issues.

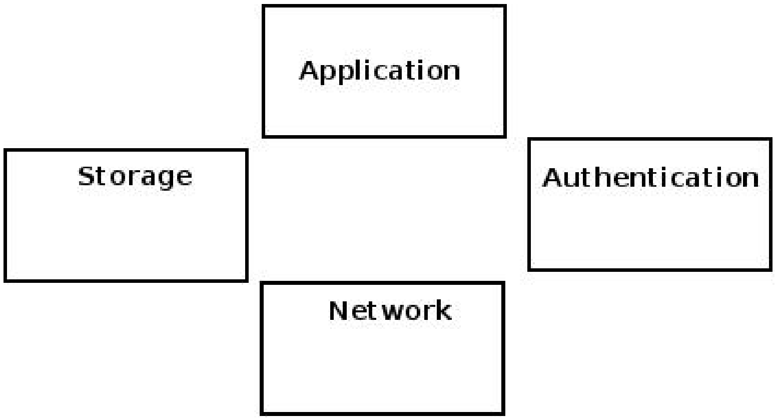

Illustration 5: Four Basic Blocks

In general, we can consider from the following aspects for the performance issues:

1. Application architecture – how many hosts are needed for the desired concurrent usage for voice/video calls and text messaging? Or intermediate servers are needed for SIP trunks or binding XMPP cluster?
2. Authentication architecture – is it necessary to have replica servers of user data for account authentication?
3. Storage architecture – the storage space is enough for the desired application?
4. Network architecture – will bandwidth be bottleneck? How shall the network be segmented for performance and security purpose?

We start with the arrangement of application servers.

Arrangement of Application Servers

We use the following diagram to specify the issues to be considered when arranging the application servers. We start with the Number Planning for Voice/Video calls via SIP. For the hosts sitting between Internet and private network, those are intended for the users to do SIP registration from the Internet. For example, numbers like 31XXXX should register on host “A01”, 32XXXX should register on “A02”, ...

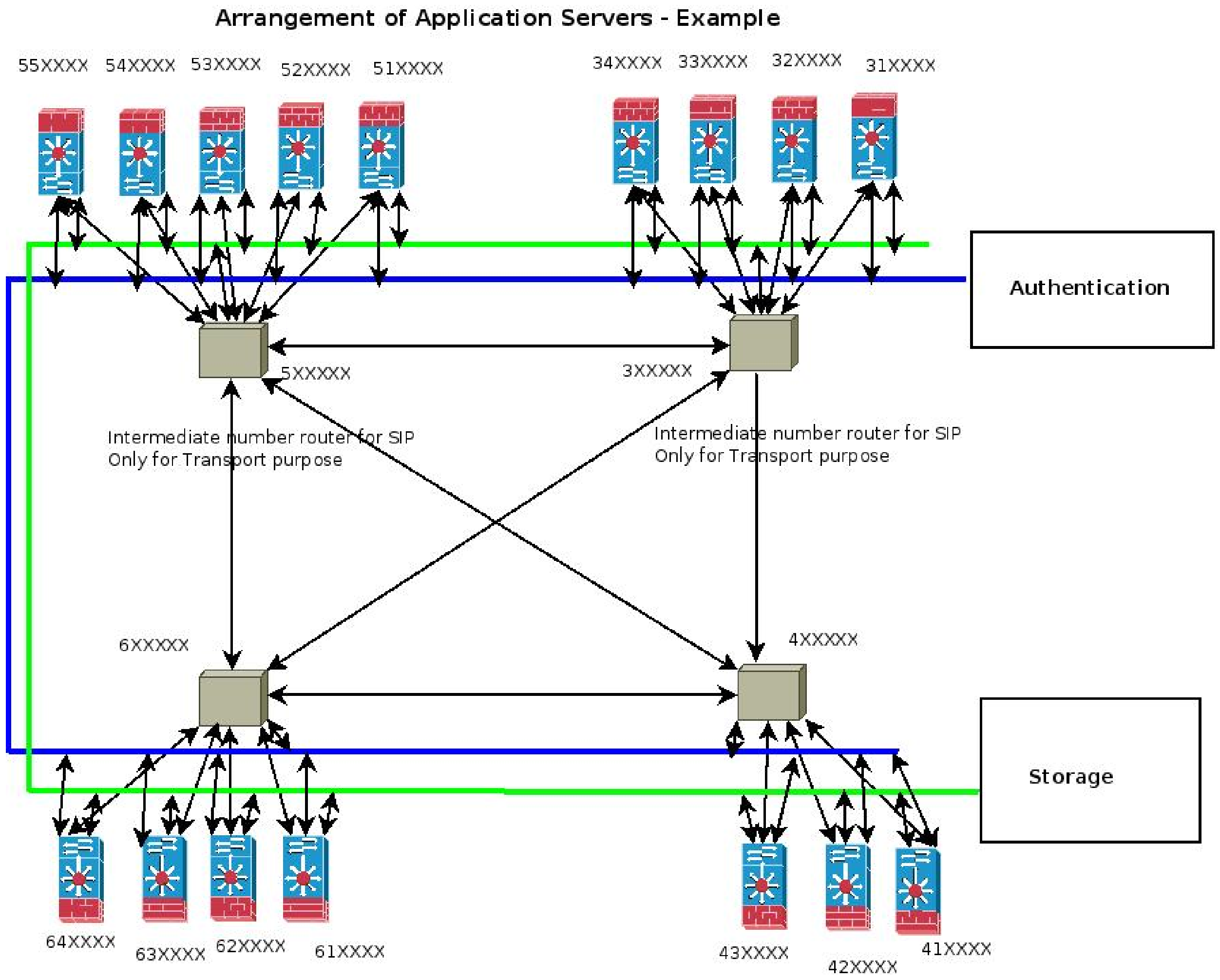
Arrangement of Application Servers - Example
55XXXX
54XXXX
53XXXX
52XXXX
51XXXX
34XXXX
33XXXX
32XXXX
31XXXX
Authentication
5XXXXX
3XXXXX
Intermediate number router for SIP
Only for Transport purpose
Intermediate number router for SIP
Only for Transport purpose
6XXXXX
4XXXXX
Storage
64XXXX
63XXXX
62XXXX
61XXXX
43XXXX
42XXXX
41XXXX

Illustration 6: Arrangement of Application Servers - Example

How should we specify the dialing rules on each host while establishing the connection via SIP trunk? If we adopt “flat” structure, we are going to have **(n-1)** entries on each host when we have **n** servers for SIP registration. This would increase the complexity of management: if we have minor changes, for example, adding a new host, it is necessary to update the configuration files on the rest of the hosts.

To minimize the changes on each host, we might introduce “intermediate nodes”. Those “intermediate nodes” are not used for SIP user registration; they are only for routing the voice/video calls from one node of its region to other nodes in that region or to other intermediate nodes. This would avoid the global changes if there is any update in a specific region.

XMPP does not need this routing scheme, but it needs to form a cluster by joining each node one by one. Between any two nodes in the cluster, it can not have the firewall to block the traffic for XMPP.

It might have the chance to use SIP trunk to connect to the other systems. The numbers associate with the dialing plan for such access should be reserved in advance. Dialing Numbers in SIP should be considered as resources: all the possible dialing numbers might be used up if the dialing plan is not devised carefully.

The reason why we would like to have multiple hosts for Voice/Video/Text is that we would like to split the loading to multiple hosts. If text messaging (XMPP) is not needed, SIP accounts (for voice and video calls) can be created in a distributed manner on each host – it only needs to establish the SIP accounts that users will come to register on that host; it does not need to have the global knowledge on all SIP accounts. But in XMPP cluster, each host needs to have the knowledge of all user accounts. Thus, a centralized server for account management is needed to avoid creating accounts on each host.

However, centralized account management could bring up another issue: the loading for user authentication might become bottleneck on this server. We will discuss this problem in the next section.

Authentication Mechanism

As mentioned in the previous section, XMPP cluster needs to have user account information populated to each node of the cluster. Thus, we would like to have centralized account management for user authentication. On each application server, it is with LDAP for account authentication. To serve the purpose, we might just prepare a LDAP server for account authentication.

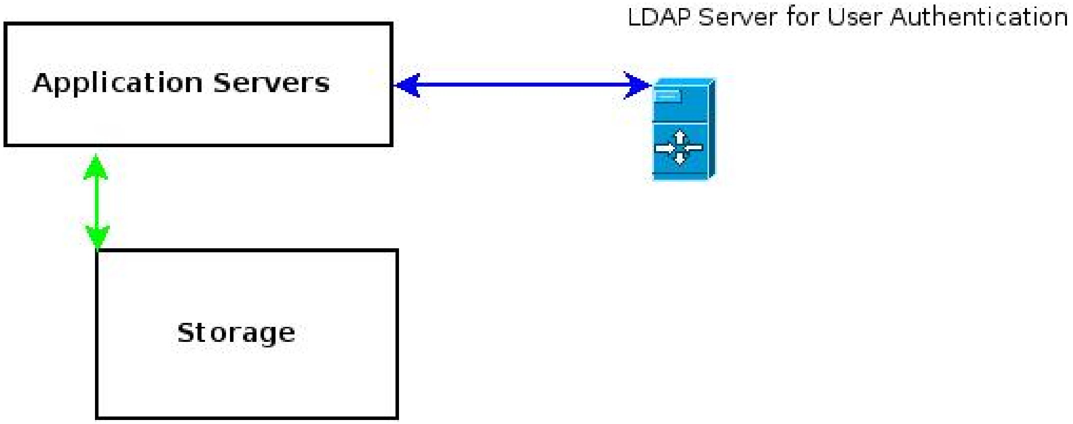

Illustration 7: Single LDAP Server for Account Authentication

Each application server will cache the result for 3600 seconds if the authentication is successful so that not every account authentication request would go to LDAP server. It still has the chance that the LDAP server reaches its limit; at this moment we might consider to expand the processing power for account authentication.

LDAP provides the mechanism to have replica servers. The replica server only accept the data queries; it does not allow updating data. Only LDAP producer will accept data update. The replica servers only take data from LDAP producer and response the queries for user authentication.

Load balancer might be used to distribute the authentication requests to different replica servers. However, load balancer is also with its limit. It is possible to let the queries of a group of application servers to a specific replica server for user authentication or using multiple load balancers.

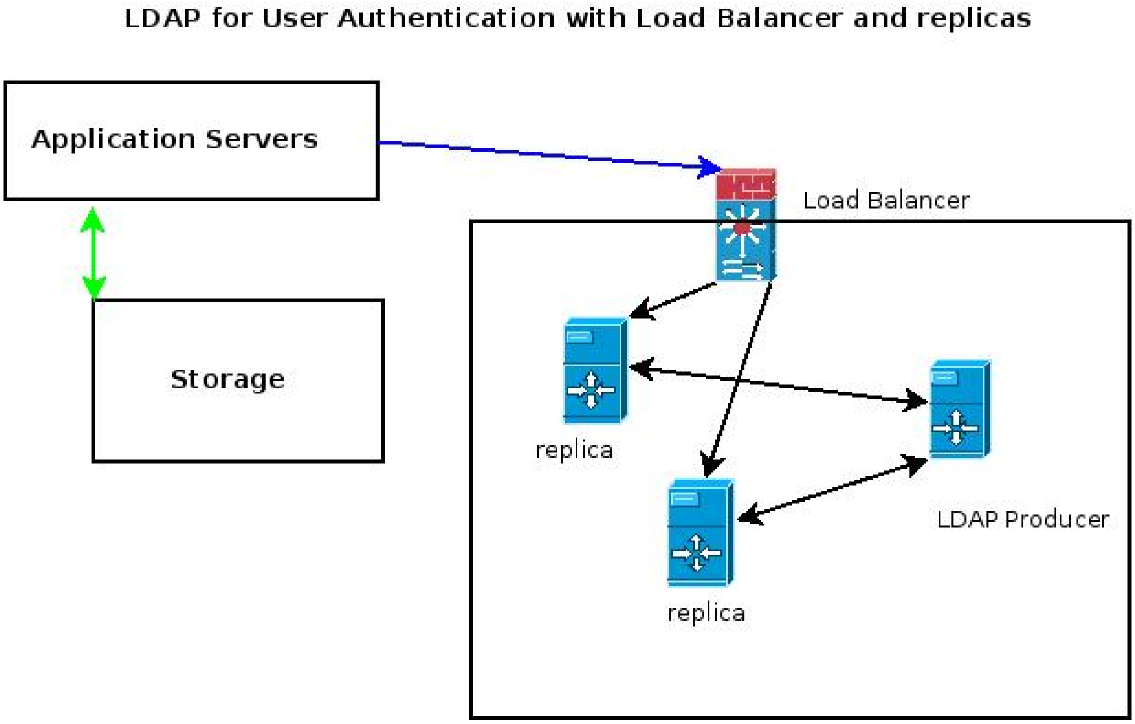

Illustration 8: Using LDAP replica servers with load balancer

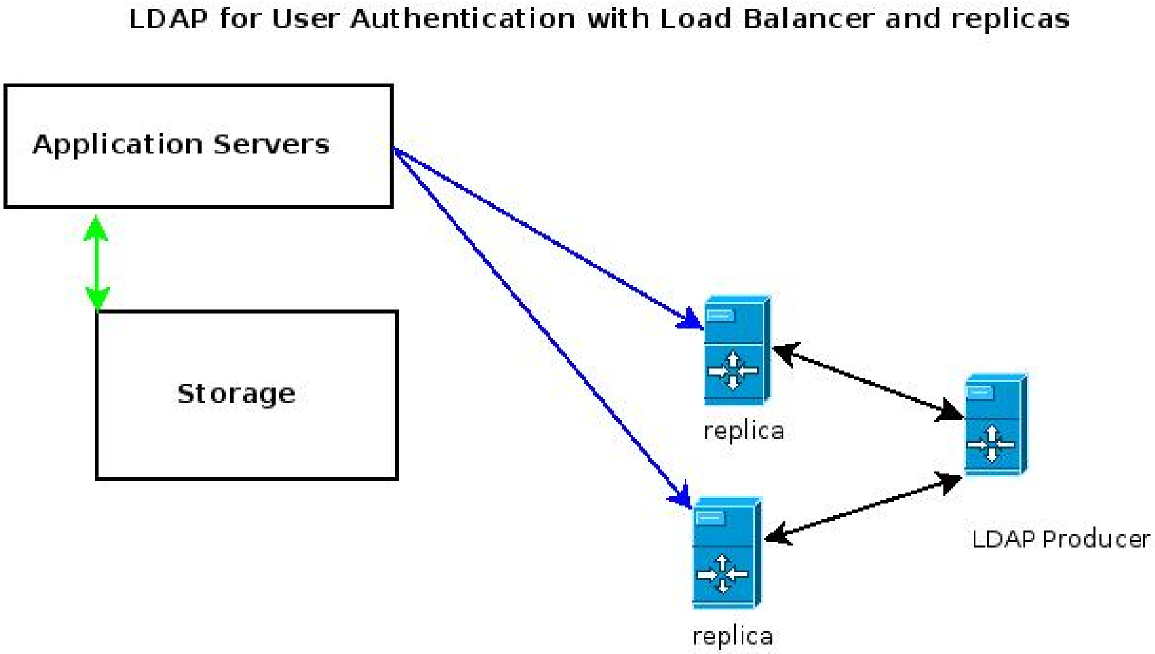
LDAP for User Authentication with Load Balancer and replicas
Application Servers
Storage
replica
replica
LDAP Producer

Illustration 9: Direct Access to Replica Servers

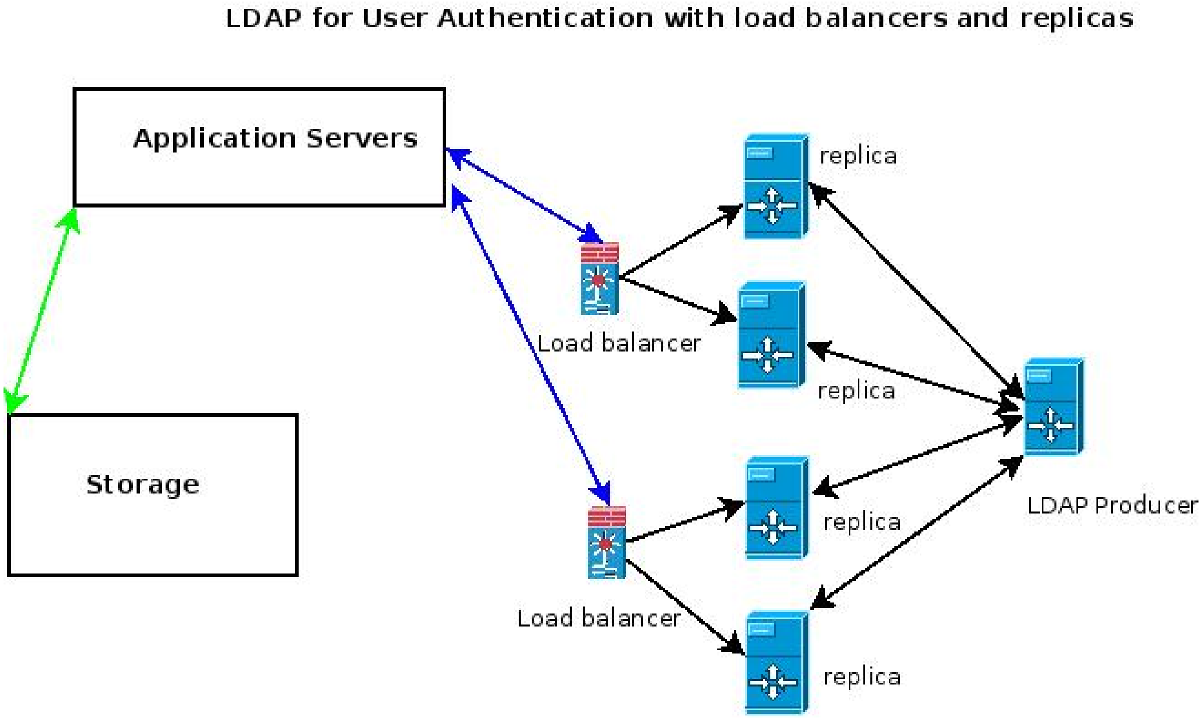

Illustration 10: Using Multiple Load Balancers with replica servers

What if single LDAP server can not hold all the user data? If single LDAP server can not hold all the user data, the schemes mentioned above will not be deployed successfully. In this case, you might consider to use “LDAP Distributed Directory Service”.

To use “LDAP Distributed Directory Service”, it is necessary to split the user data into different subtrees, put the subtree data into other servers, and create referral on the main server database. We will not go through all the details here. Instead, we just give high level description here. When the main server gets the query, it returns the referral information if the desired data is located in other servers. Then, the client issues another query to the server that referred by the main server. The whole process is called “chase referral” in LDAP terminology. The application servers are turning on “chase referral” by default so that they will go after the referral automatically.

The major drawback of “chase referral” is that the client needs to issue the data query twice – the first time to the main server to get referral, and the second time to the server where the date is situated.

Thus, those servers with subtree data shall also be accessible by the application servers directly.

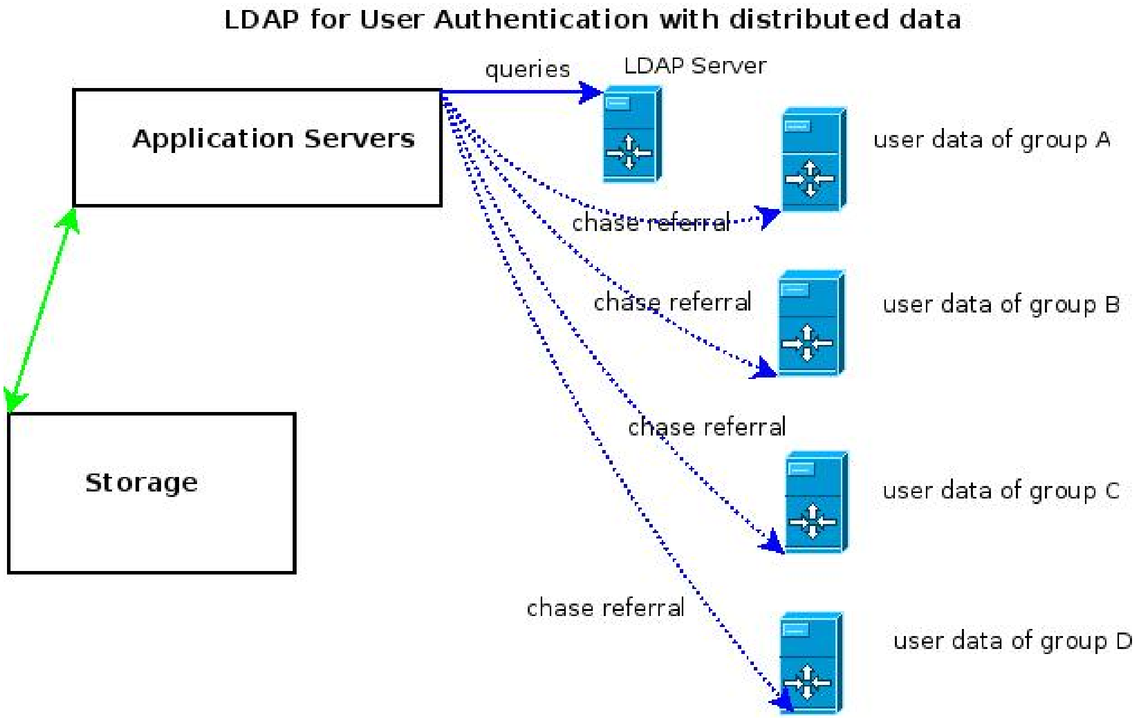

Illustration 11: LDAP Distributed Directory Service

In general, user data in LDAP server should not occupy too much space. If possible, it is desirable to consider put all data into one server and do replicas for improving performance.

Storage System

The deployment of the storage system could be quite different in response to different use patterns or business requirements. The application servers use extra storage space in two areas: **file sharing in XMPP** and **recording voice/text conversation for audit purpose**. At the moment of writing this document, video recording is not done on server side – it is done by using client programs.

The following factors shall be considered while selecting storage solutions:

1. the storage space
2. the transfer data rate
3. the maximal number of files allowed in the file system

The application servers can use external storage space via Samba or NFS. Thus, the solution can be as simple as using single File server or NAS (network attached storage):

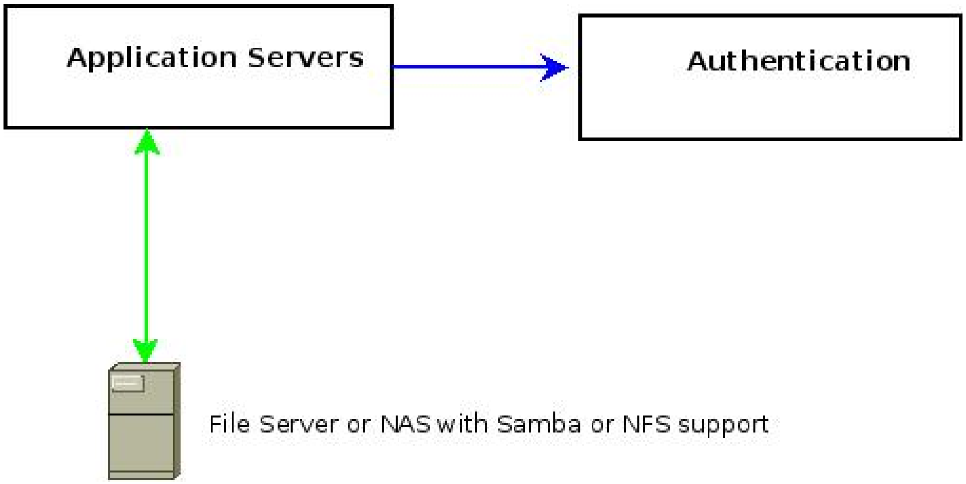

Illustration 12: single File Server or NAS

If the storage space is not an issue, the bandwidth for data transfer is in concern, multiple Ethernet interfaces would be a good choice on File server or NAS. As mentioned earlier, the storage system in mainly used for file sharing in XMPP or recording voice/text conversations. The usage of “file sharing” in XMPP is that one user is doing **HTTP Upload** to the storage and the other end of user just fetches the file; recording voice/text conversation is to combining RTP streams or text messages from both sides into a single file. Thus, the bandwidth needed for the storage system should not be less than the bandwidth of the application servers facing the user side.

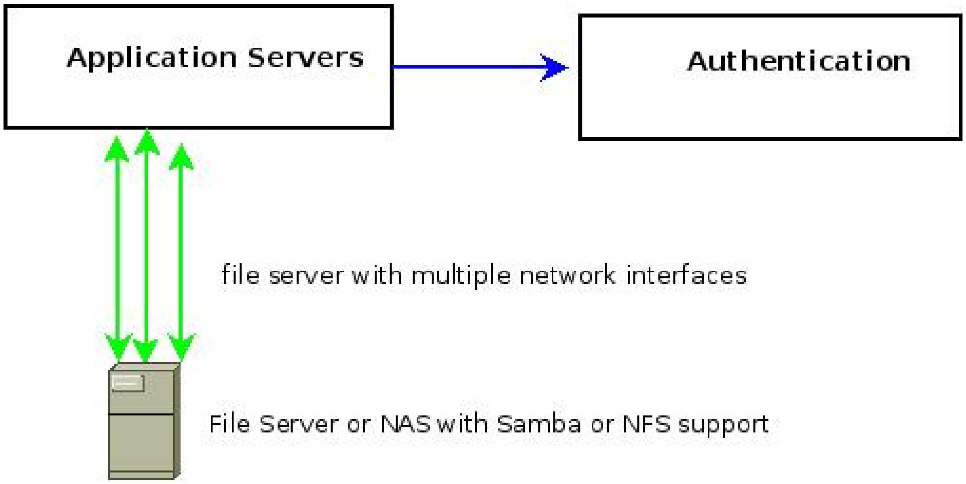
Application Servers
Authentication
file server with multiple network interfaces
File Server or NAS with Samba or NFS support

Illustration 13: Multiple Ethernet Interfaces for File Server or NAS

Could it be possible by using several independent file servers? For “file transfer” in XMPP, it means some of the application servers have to be accessible from Internet without using VPN; or those application servers serve as “file sharing proxies” shall be open to the all users from VPN. In general, we do not recommend those approaches. But if you only allow users registering on the same group of application servers (mounting the same storage) to do file sharing, several independent file servers can be taken into account.

If the bandwidth provided by single file server does not meet the demand, you might consider using Samba Cluster.

Samba cluster uses “cluster file system”, such as OCFS2 or GFS2, to manage “hard drive devices” from private network; usually it is via iSCSI or AoE. And those Samba nodes are with one side facing the users. The users will find out the content is the same by accessing from any one of those Samba nodes.

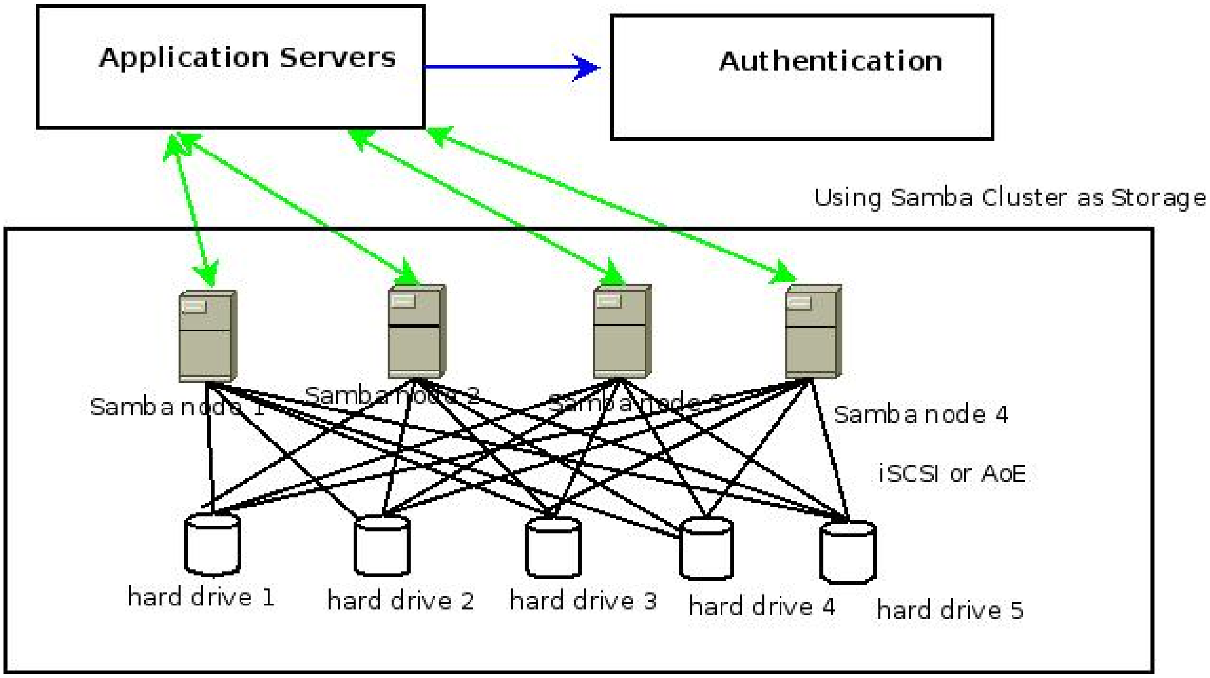

Illustration 14: Samba Clustering

With the bandwidth requirement you have , you might determine how many Samba nodes you should deploy.

The limitation of Samba Clustering usually comes from “cluster file system”. Usually it is with better performance when the scale is smaller. The hard drives designated in the diagram above are not just “hard drives”; a single hard drive we mean in daily life can not have network capability. They are usually the “virtual hard drives” (for example, iSCSI targets provided by iSCSI portal) on NAS. It might not be easy to manage.

The other choice is Hadoop HDFS with its NFS 3 gateway.

Hadoop can be used for parallel processing, but we only use its HDFS as storage system.

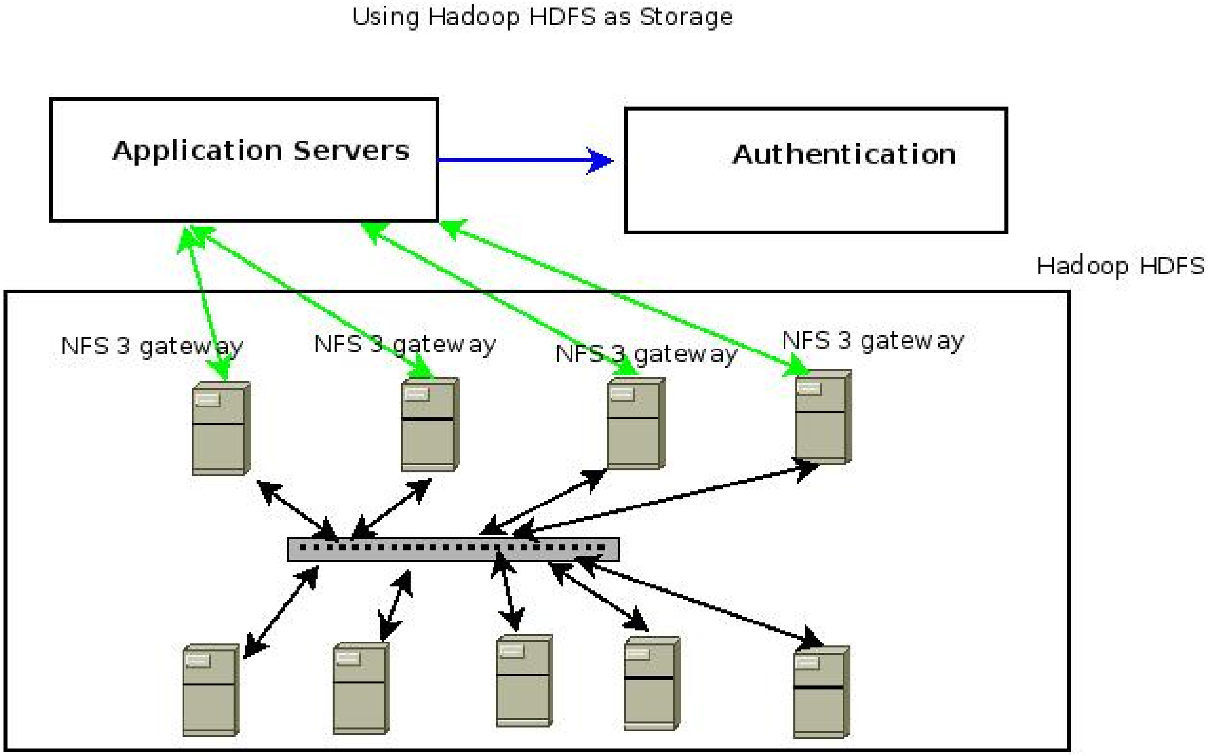

Illustration 15: Using Hadoop HDFS as storage system

Usually, it does not allow modification on the file content (it only allows “appending” the file) while accessing via Hadoop HDFS NFS 3 gateway. For “ File Sharing” in XMPP, it is “write once and leave it there for reading”. Thus, it is with less problem to access via NFS 3.

If constantly writing (appending) into a file via Hadoop NFS 3 gateway, the content of the file will be intact while fetching through those Hadoop commands; but reading from NFS 3 gateway might get empty content. Thus, it does not have too much freedom like other NFS or Samba system.

Network Planning

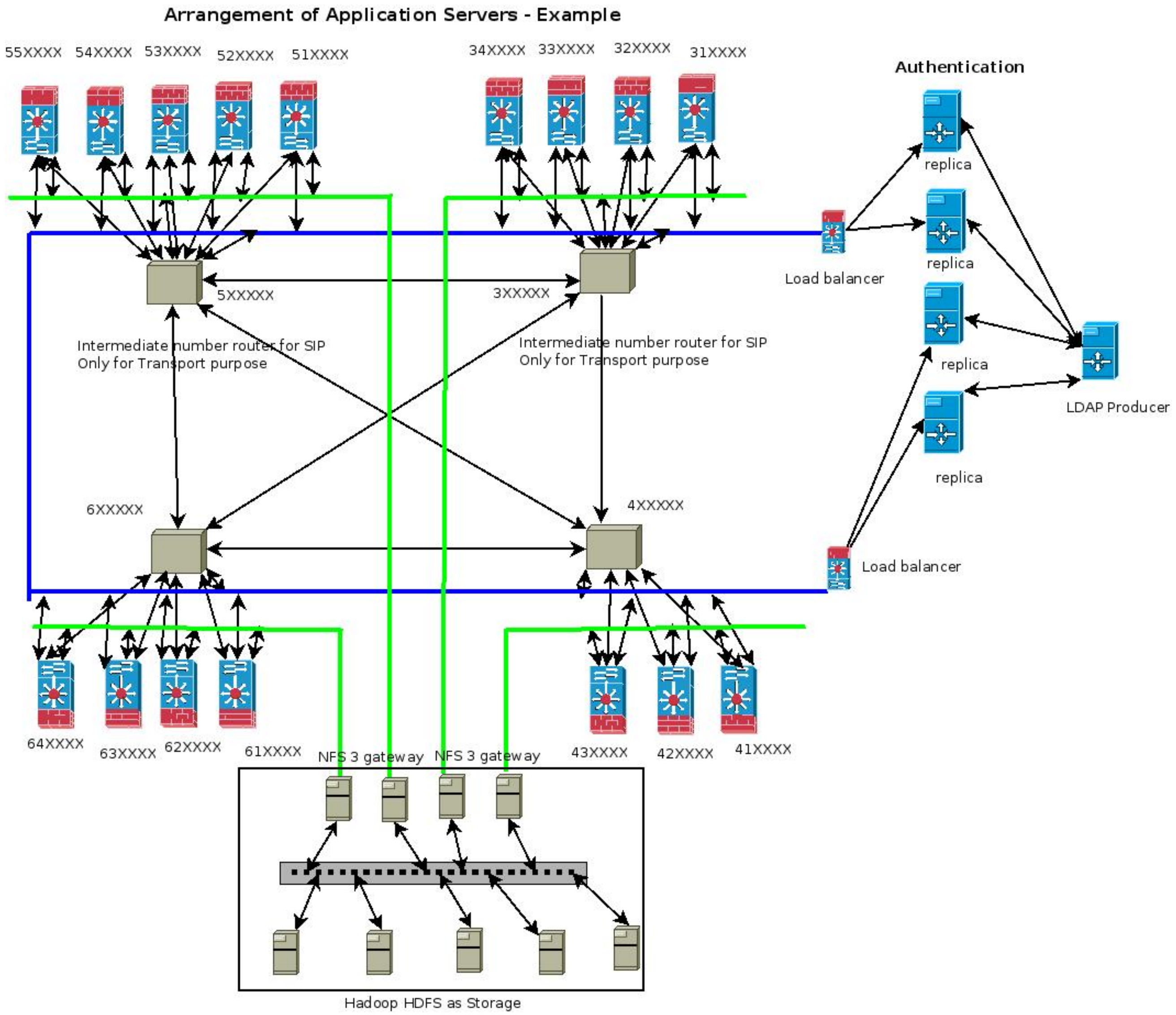

Illustration 16: Deployment Example

The diagram above is an example by just putting everything so far we have together. And do not forget that XMPP cluster needs to have all nodes to see each other. In the diagram, each application server is at least with two Ethernet interfaces: one is connected to Internet and the other is to the private network. But whether authentication and storage access should use the same Ethernet interface on each application server? If extra Ethernet interface is needed, then each application server will have 3 Ethernet interfaces.

The network planning is not only to identify the network bottleneck in advance, but also to consider the ease of trouble shooting when one of the network equipments is broken.

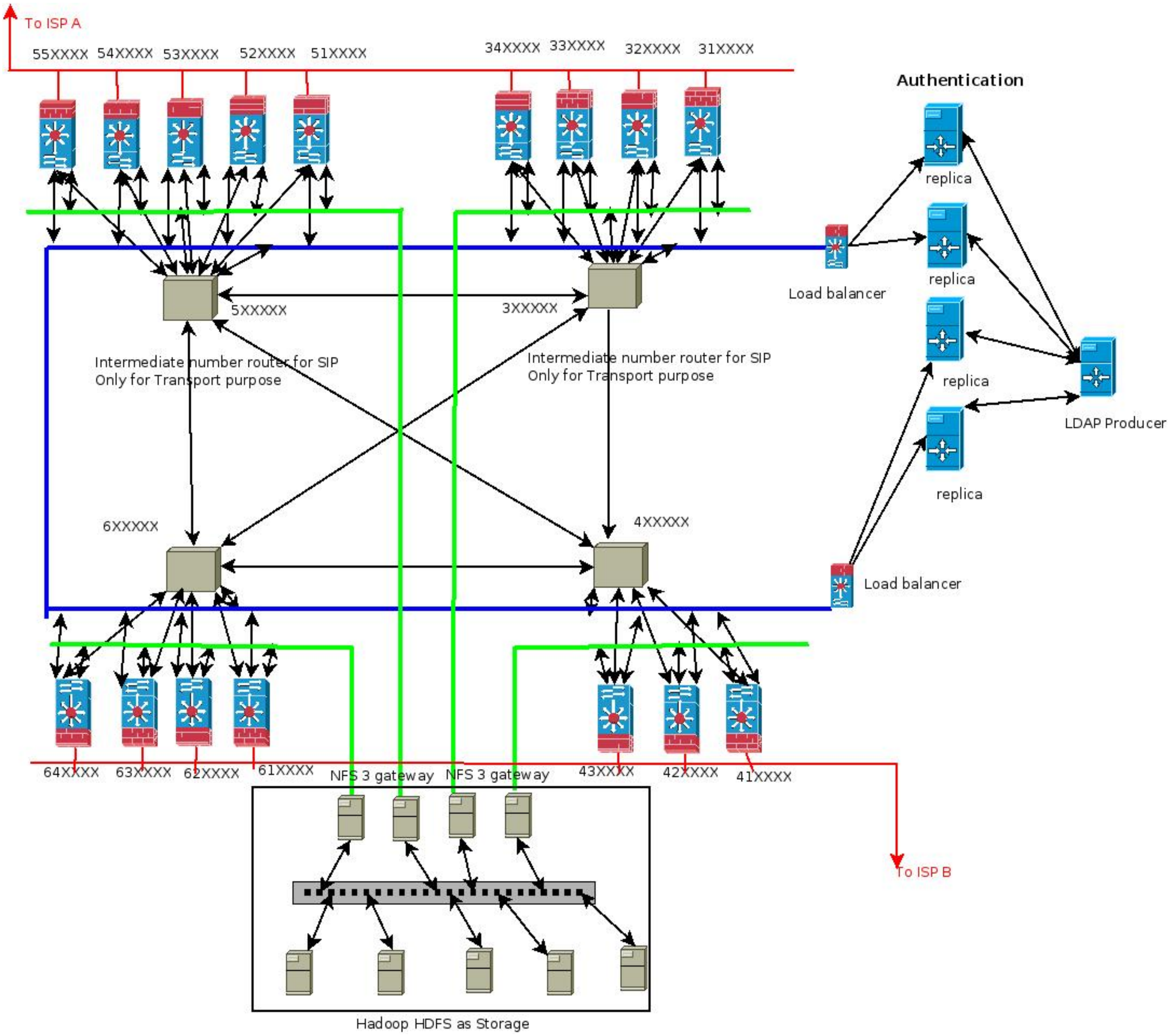

Illustration 17: Using Multiple ISPs for Internet Connection

The diagram above is to use multiple Internet Service Providers (ISPs) for Internet connections.

In general, we would suggest put all routers and switches on the diagram, and identify the port(s) and IP address(es) for each server. If static routing is used, spell out the routing entries should be added on each server. Please note: for those application servers with at least two Ethernet interfaces, it needs more planning to put their Ethernet interfaces connecting to the private network into different subnets. Their default gateway settings shall be the gateway settings provided by the ISP(s). Thus, the routing entries to different subnets need to be added one by one while static routing is used.

If firewalls are used in the private network, they should be identified earlier as well. Voice/Video calls are not only involved with SIP; the associated RTP streams should also be taken into account. The failure of RTP to pass through the firewalls will cause no voice or no video on the other hand.