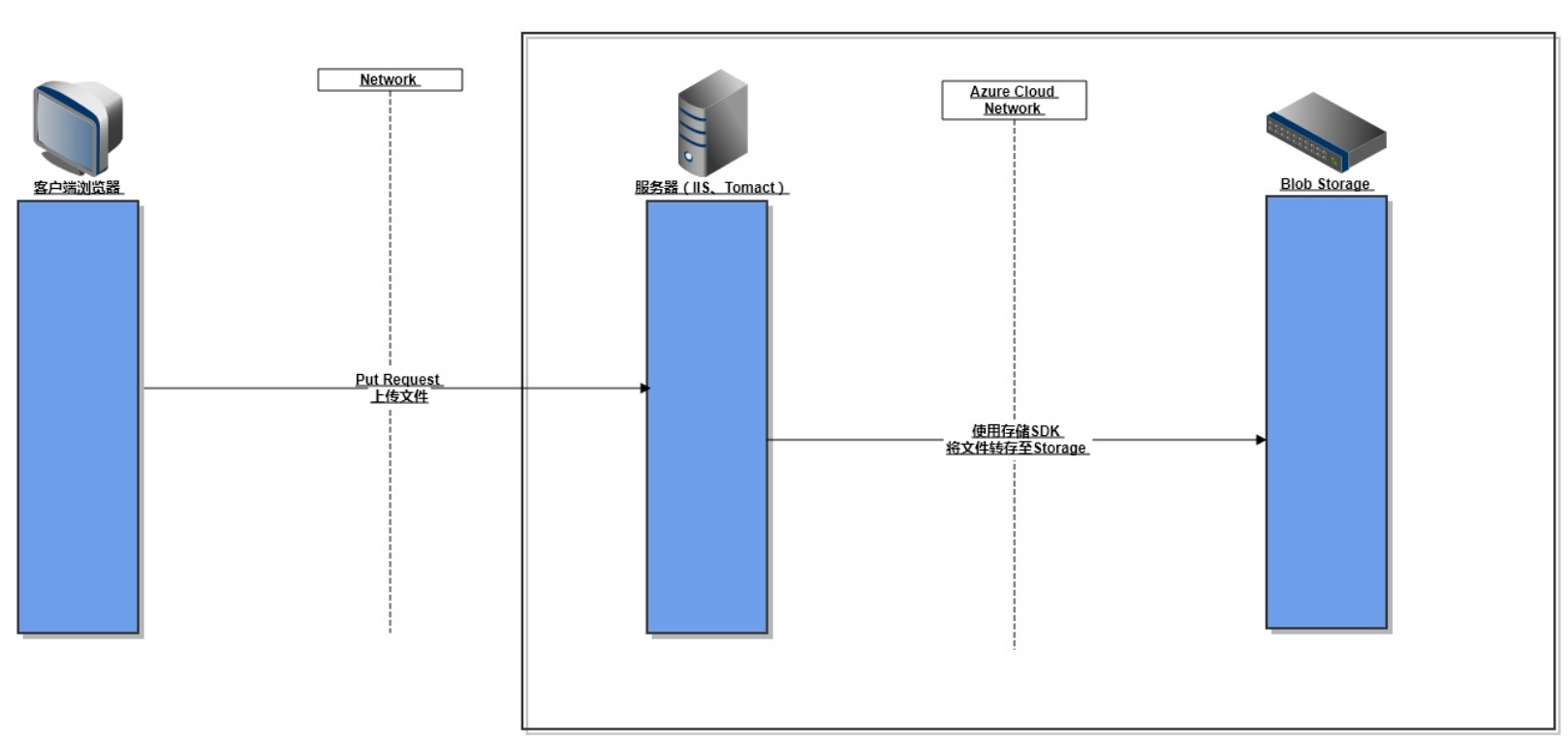
**使用Javascript实现跨域上传文件到存储**

用户在Azure上构建B/S程序时，有很多场景需要存储海量的文件，比如图片、文档及视频等，按照传统的方式，很多用户会将这些文件存储到宿主服务器上，比如基于IIS网站可能会考虑将这些文件存放在虚拟目录下某个文件夹中。不过目前大部分的Azure计算服务在一定程度都有容量限制，比如虚拟机可附加的磁盘数量和单块磁盘容量也只能支撑几十TB的总容量。PaaS的计算服务，如Web App、Cloud Service、Service Fabric等因无法附加磁盘，所支撑的存储容量更是少之有少，往往支持到几十或几百GB。

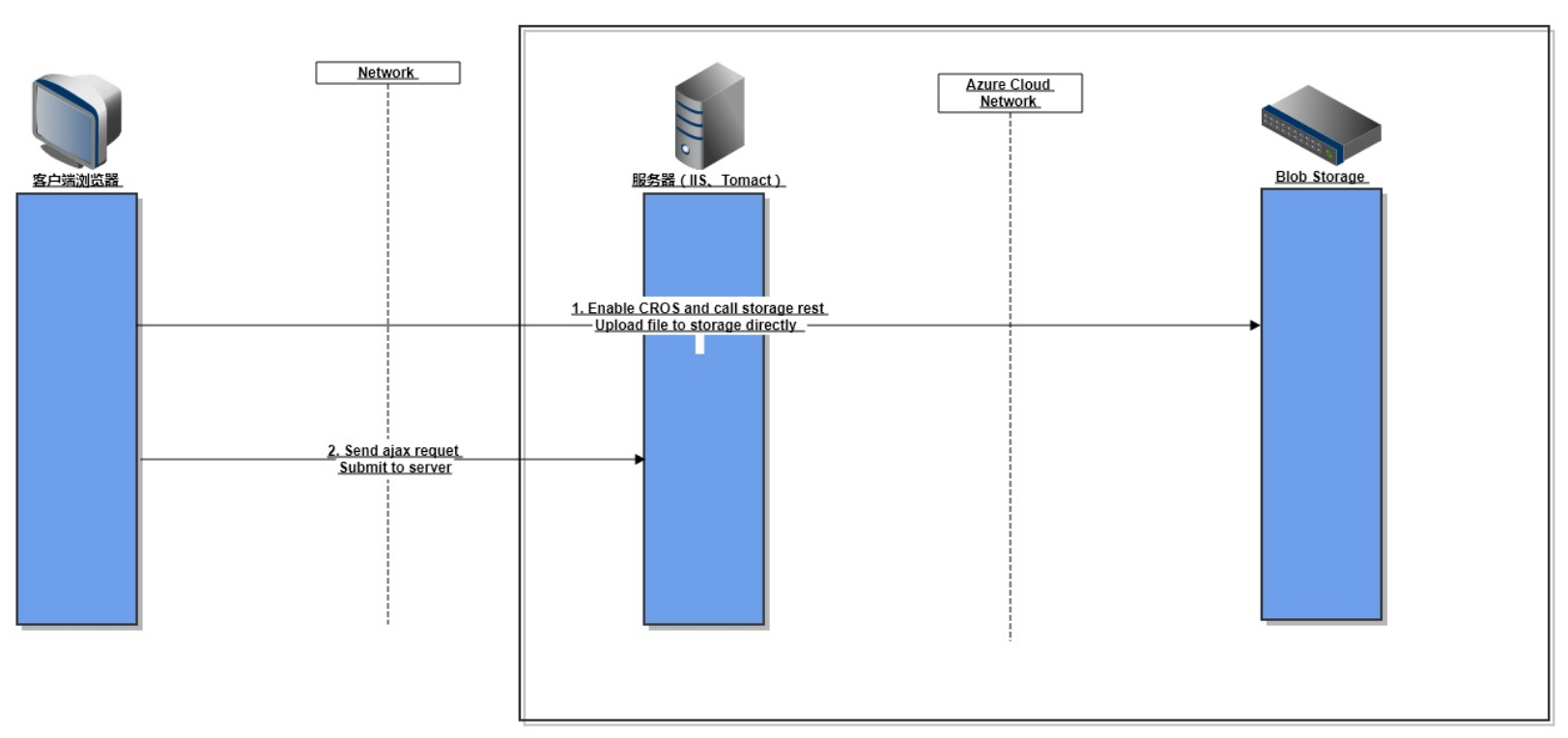
因此，对于云端B/S应用产生的文件存储，我们急需一种有效的存储方式。从现有的服务及适用场景来看，将应用文件托管到Azure Blob Storage服务，是最为有效的方式。存储单个账户规模可以支撑500TB容量，通过配置多个账户，可以支撑应用文件的海量存储，达到线性扩展的目的。

那么，B/S应用如何将文件上传到存储服务呢？一般有两种方案：

* 方式1：使用传统逻辑将文件从浏览器端（Client）上传到虚拟机（Server）,然后在调用存储接口，将文件从虚拟机（Server）上传到存储（Azure Storage）



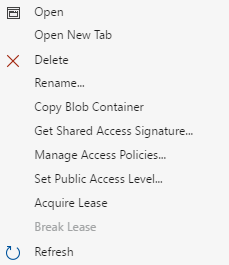
* 方式2：将文件从浏览器端（Client）直接跨域上传到存储（Azure Storage）,同时在发起Ajax请求到虚拟机（Server）,在虚拟机（Server）接收上传的文件信息，进行后续逻辑处理，比如写入数据库等



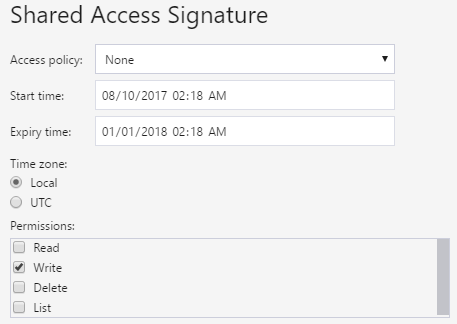
第一种方式略显笨拙，缺点比较明显，涉及中转过程，但不涉及跨域，不太适合互联网类型应用。第二种方式基于Ajax技术，不涉及中转，但需要了解存储REST接口、跨域访问等知识，实现难度略大一些，不过是最佳的方案选择。

下面我们将介绍第二种方案的实现，在此之前我们需要掌握一些基本的知识，参考如下：

1. 存储共享访问签名（SAS）
2. 下载安装：[Azure Storage Explorer](http://storageexplorer.com/)
3. 连接目标存储账户，在目标“存储容器”上点击右键，在弹出的菜单项中选择“Get Shared Access Signature…”:



1. 上传文件需要写入权限，如果需要持续上传，可以调整过期时间，示例如下：

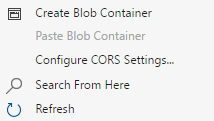


1. 点击“Create”，在弹出的界面上我们即可获取SAS签名URL。Query String即是签名字符串，这个签名字符串包含操作权限、有效时间等，如下：

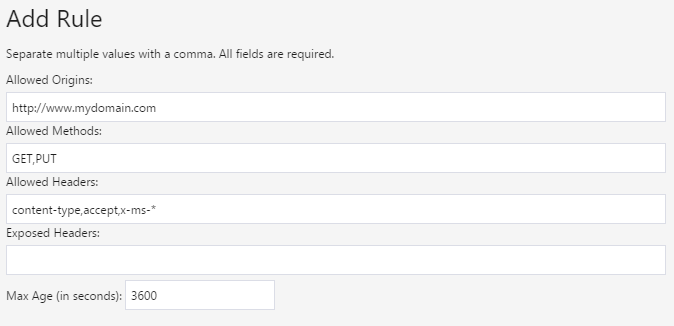


[https://devstoragerm.blob.core.chinacloudapi.cn/files**?st=2017-08-09T10%3A18%3A00Z&se=2017-12-31T10%3A18%3A00Z&sp=w&sv=2016-05-31&sr=c&sig=kupC7DiU0vaTMqY3DZ3sbl96MV21efmXwd8yDix%2BJ7E%3D**](https://devstoragerm.blob.core.chinacloudapi.cn/files?st=2017-08-09T10%3A18%3A00Z&se=2017-12-31T10%3A18%3A00Z&sp=w&sv=2016-05-31&sr=c&sig=kupC7DiU0vaTMqY3DZ3sbl96MV21efmXwd8yDix%2BJ7E%3D)

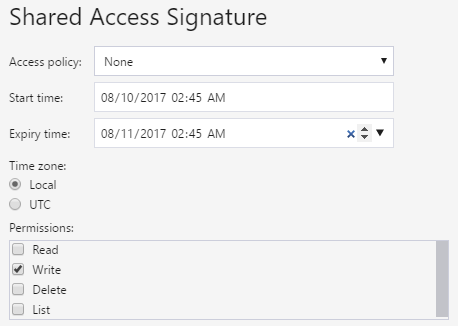
1. 更多参考资料，如下：
2. [使用共享访问签名委托访问权限](http://msdn.microsoft.com/ZH-CN/library/azure/ee395415.aspx)
3. [Shared Access Signatures, Part 1: Understanding the SAS Model](http://azure.microsoft.com/en-us/documentation/articles/storage-dotnet-shared-access-signature-part-1/)
4. [Shared Access Signatures, Part 2: Create and Use a SAS with the Blob Service](http://azure.microsoft.com/en-us/documentation/articles/storage-dotnet-shared-access-signature-part-2/)
5. [Introducing Table SAS (Shared Access Signature), Queue SAS and update to Blob SAS](http://blogs.msdn.com/b/windowsazurestorage/archive/2012/06/12/introducing-table-sas-shared-access-signature-queue-sas-and-update-to-blob-sas.aspx)
6. [创建和使用共享访问签名(C#示例)](http://msdn.microsoft.com/ZH-CN/library/azure/jj721951.aspx)
7. [Controlling Access to Azure Blob Containers with Java(JAVA共享签名)](https://msdn.microsoft.com/en-us/library/azure/dn246928.aspx)
8. 存储跨域设置(CROS)
9. 下载安装：[Azure Storage Explorer](http://storageexplorer.com/)
10. 连接目标存储账户，在“Blob Containers”上点击右键，在弹出的菜单项中选择“Configure CORS Settings…”:



1. 在上一步操作弹出的界面，点击“Add”，配置CROS规则，示例如下：

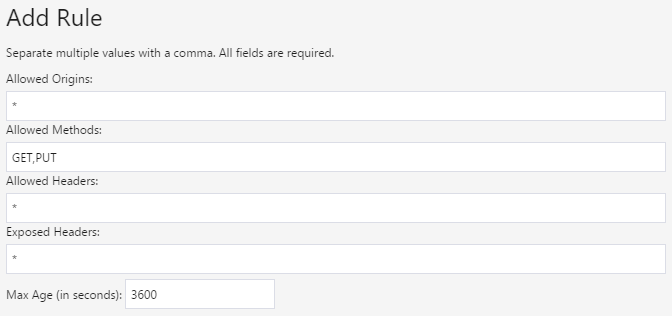


1. 更多关于存储的CROS设置参考：[Cross-Origin Resource Sharing (CORS) Support for the Azure Storage Services](https://docs.microsoft.com/en-us/rest/api/storageservices/cross-origin-resource-sharing--cors--support-for-the-azure-storage-services)
2. 存储REST接口
3. [存储服务 REST API 参考](https://msdn.microsoft.com/zh-cn/library/azure/dd179355.aspx)
4. [PutBlob](https://docs.microsoft.com/en-us/rest/api/storageservices/Put-Blob?redirectedfrom=MSDN):  适合上传小文件的REST API
5. [PutBlock](https://docs.microsoft.com/en-us/rest/api/storageservices/operations-on-block-blobs): 分块上传，可以实现大文件上传、进度控制、断点续传的REST API
6. 完整的示例：
7. 下载代码示例：<https://github.com/hello-azure/azure-storage-javascript>
8. 在要上传的容器上，创建一个“写”权限签名URL:

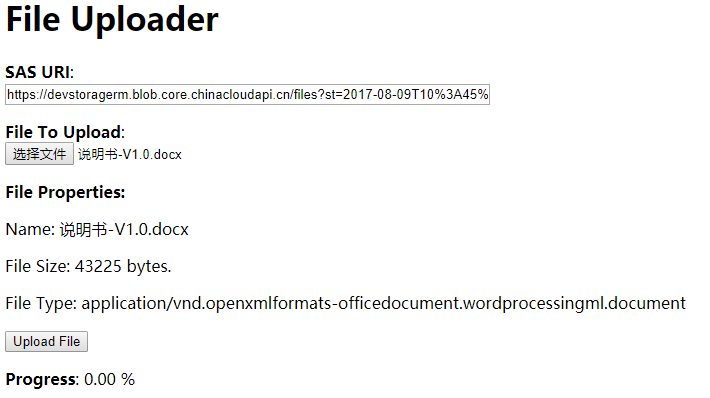


签名URL示例：<https://devstoragerm.blob.core.chinacloudapi.cn/files?st=2017-08-09T10%3A45%3A00Z&se=2017-08-10T10%3A45%3A00Z&sp=w&sv=2016-05-31&sr=c&sig=reFh8jikl%2B2hOVhJ5SORPkItoFN6Y8bbq%2FeYgKn6QsI%3D>

1. 配置CROS，部署到生产环境，请限制为响应允许的Origins、Methods、Headers，这里我们在本地测试，使用\*提供了“宽范围”的CROS设置：



1. 使用支持H5的浏览器打开test.html，将签名URL粘贴到“SAS URI”,选择目标文件后，点击“Upload File”,完成上传。过程显示如下:



上传完成：



查看容器：



1. 其它相关参考资料：
2. [New Azure Storage JavaScript client library for browsers – Preview](https://blogs.msdn.microsoft.com/windowsazurestorage/2017/03/07/new-azure-storage-javascript-client-library-for-browsers-preview/) (MSDN)
3. [Reliable Uploads to Blob Storage via an HTML5 Control](https://docs.microsoft.com/zh-cn/rest/api/storageservices/reliable-uploads-to-blob-storage-via-an-html5-control) (MSDN)
4. [Uploading Large Files in Windows Azure Blob Storage Using Shared Access Signature, HTML, and JavaScript](https://stackoverflow.com/questions/19935404/upload-largefile-to-azure-blob-using-javascript) (Stackoverflow)