File analysis tool Requirement analysis

目录

1. Background

Amazon Device OS team is using Artifactory to host services (https://storage.labcollab.net and https://storage-cnn.labcollab.net) to share project files with external parties, e.g Contract Manufactory, ODM vendors, per each device project request.

Artifactory admin need ensure every project shared files compliant to Amazon security policies, avoid information leakage (refer https://policy.amazon.com/policy/95 for reference) while project teams using the temp share repos/folders. By now, this is a manual operation workload and cost much effort and increasing fast along projects. This doc is proposing a new automation way to timely running analysis against the file content , to identify violations and take prompt actions to avoid unwanted information leaking cases.

Initially, the idea is to run the detection at two levels and raise corresponding warning/errors to gain project lead/TPM awareness before further actions.

Default level - applied to all repos/folders, e.g. all source code with Amazon Intellectual Property are not allowed uploaded to any Artifactory repo /folder shared with third party.

Project level - TPM or Tech Lead to define what kind of content are absolutely violate.

2. Requirement analysis

2.1. User requirement

File analysis tool is a tool for analyzing files in the Artifactory repo/folders to check if any violations. When a user uploads a file to Artifactory, the tool will "filtering" all uploaded files and to analyze if the files are compliant. If the outcome doesn't show any violations under current detection rules, then the files will be shared to external parties as expected; Otherwise it will be intercepted on the contrary(can't be shared to anyone).

During the analysis process, this tool will send some emails to users about the progress of file analysis. File analysis tool also contains a web page. The web page's link will be contained in each email, to help users track the analysis progress and get analysis histories in the web page.

At the same time, If a non-compliant file was unloaded to Artifactory, Admin will receive a email too. Admin can also login to the web application to get analysis logs.

User's and Admin's Requirements are as following:

Artifactory user:

- **Upload files**
  User uploads files to the Artifactory. After that, file's permission will be locked. So others can't visit this file.
  In order not to affect the user experience, the analysis needs to be completed within 1h
- **Define analysis rules**
  TPM or Tech Lead to define what kind of content are absolutely violate.
- **Notification of analysis process**
  User need to know the progress of file analysis.
  Notification includes "file is ready to be analyzed" and "file analysis is completed".
  If file is compliant, file's permission will be unlocked and it can be visited by others. If file is non-compliant, others can't visit it and email notification will contains why it is non-compliant.
- **Web page**
  To get more details and histories, user can login to the web page.

Artifactory admin:

- **Notification of non-compliant files upload behavior**

If a user uploads a non-compliant file, Admin should get a email notification.

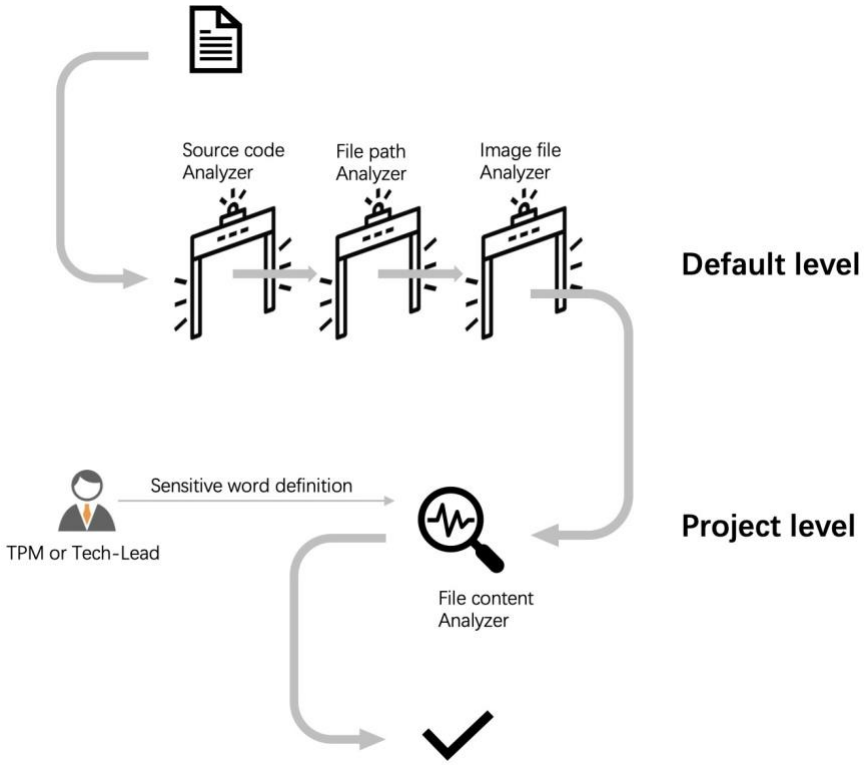
- **Web page**

The tool need to record all upload and analysis information. These information can be find in the web page.

2.2. Detection schema

This service attempts to establish two analysis levels to check the security compliance of the file content. One is the default level, which makes specific default information security analysis for file content. Cover all repos by default; Another level is user-defined, project-level file analysis based on sensitive word detection.

At this stage ,File analysis tool provides a rough file analysis. These analyzers can be expanded in the future.



2.2.1. Default level

Default level is a basic level analysis for all repos in Artifactory. This process is aim for some konwen non-compliance situations. The default level analysis is for three aspects, "Unpublished source code" , "File path" and "Image file".

Unpublished code analysis:

Code analysis is the analysis of source code in Artifactory. Some unpublished Amazon Source Code should not be uploaded to Artifactory, so this analysis is performed.

File analysis tool will build a DB includes all known Amazon Source Code's SHA256 value(Such as FireOS 8's fire-prop source code). Code analysis will use the SHA256 matching method to determine whether the content of the file comes from an Amazon Source Code Repo. When a user uploads a text file, user-file's SHA256 will be used to match in the SHA256 DB. If the same SHA256 value as the user file is found in DB, the file must come from Amazon Source Code.

File analysis tool will maintain a dynamic "source code SHA256 DB", which is continuously expanded according to the changes in the unpublished code Repo in code.amazon.com and gerrit6.

File path analysis:

Different vendors of Artifactory are allowed to access different projects. Therefore, if files from other project are uploaded to the vendor's project directory, it is not compliant.

For example, there is vendor-A and vendor-B. vendor-A allows access to project-A's directory, and vendor-B allows access to project-B's directory. If a file of project-A are uploaded into the directory of project-B, vendor-B will access files that do not belong to its own scope of authority.

Analysis for this situation: The folder name of each project can be considered as the name of the project. If the names of other projects appear in the name and content of a file, it is judged that this file should probably not be uploaded to this folder. File analysis tool will send a alarm information to the project owner to make a manual judgment.

Image file analysis:

Image files need to be uploaded to another service called KBITS. So, Image files should not be uploaded to Artifactory.

Image file‘s name has some stable patterns. For example, most of Image file's name match the pattern : release-<project name>-*.tgz. And some other Image Files have very clear patterns too.

So File analysis tool can judge whether the file is a Image based on the filename.

2.2.2. Project level

Project level is used to analyze files that can read text content, using the method of sensitive word analysis. If sensitive words appear in the content of the file, the file is not compliant.

Sensitive words are provided by TPM or Tech Lead of the project. TPM and Tech Lead should provide a configuration file in the specified format which contains all Sensitive words. File analysis tool will make the Project level analysis base on this file.

During the development process of a project, TPM and Tech leader can continuously maintain and expand the sensitive words of the project.

TPM or Tech Lead can define their sensitive word by follow steps:

1. Create a sub-ticket for a specified master jira ticket in https://issues.labcollab.net/.
2. Add sensitive word in sub-ticket's description by this format:
   [Path in Artifactory]:[sensitive word1],[sensitive word2],[sensitive word3],.....
   For example : vendor_test/test:xxx,xxx,xxx,xxx. (If your sensitive word contains "," ,you need to use "/," to ensure correct parsing).
3. TPM or Tech Lead start ticket progress
4. File analysis tool will scan this sub-ticket and parse out the sensitive words in it.
   If sensitive words can be parsed, the tool will put the sensitive words into the “file analysis process” and add the parsing result to the sub-ticket as a comment.
   If sensitive words can't be parsed,the tool will add error comment in ticket.
5. File analysis tool close the ticket. TPM or Tech lead can reopen this ticket and the tool will continue this process.

2.3. Tool Analysis Capability Requirement

On some dates, more than 50G files are uploaded to Artifactory. Considering that there are multiple Artifactory instances. File analysis tool need to be able to analyze 200G+ of files per day(details). So tool involve a large number of IO operations and computing behaviors such as file download and file parsing.

In order to maximize the utilization of server resources, use concurrent file analyze and file download scheme. Tool can handle multiple download and file analysis tasks simultaneously.

3. Time Line

https://playbook2.amazon.com/project/790319

4. High level Design

4.1. Architecture design

4.1.1. Component design

File analysis tool has many different designs. These differences mainly focus on two aspects:

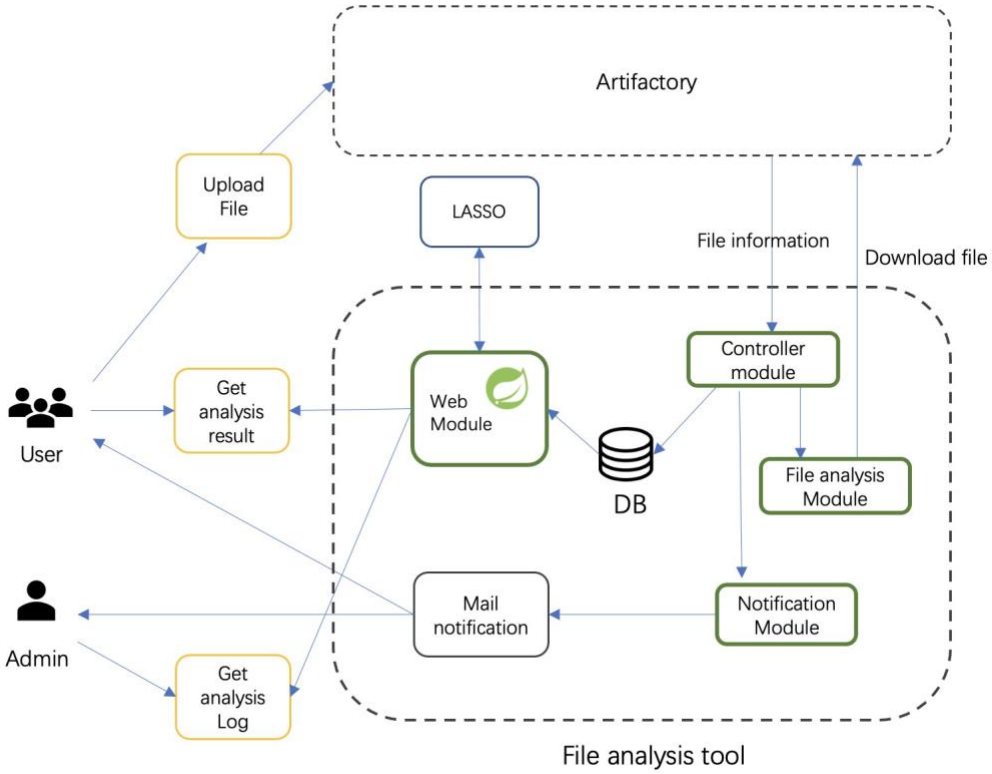
-
- Tool runs in a stand alone server or in the server that Artifactory is running in. User upload files to File analysis tool firstly or to Artifactory directly.

After comparison, the tool will work as below:

Aritfactory user upload a file to Aritfactory instance and the tool get notification from Artifactory that a new file is uploaded. Then the tool download the file, analyze file compliance and provide analysis result and notifications.

For more details, please visit Comparison of file analysis tool's upload plans

According to this design, the architecture of the entire service system is as follows:



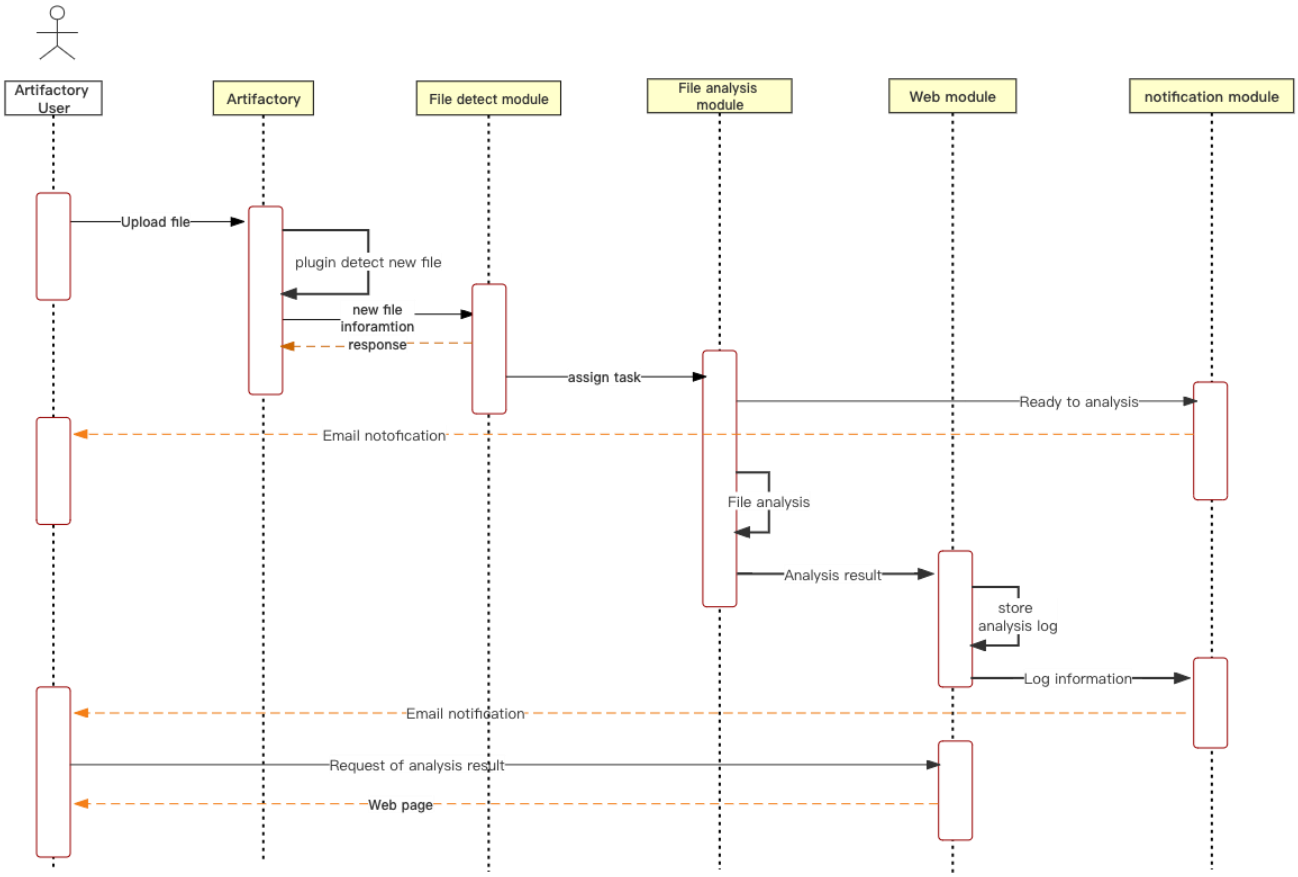
There are four main modules in file analysis tool: Controller module, File analysis Module, Web module and Notification Module.

Controller module can receive new file information from Artifactory. Each new file's information will be assigned as a task to "File analysis module".

“File analysis module” can handle multiple tasks at the same time. It's responsible for file download, file decompress and file analysis. After "file analysis module" has completed all these process, it will send analysis result back to "Controller module". Controller module will write the analysis result into the database.

As the file analysis module is running, “Notification module” will send some email notification about file analysis progress to users and email notification about non-compliant file to admin. In addition, there will be a link in the email. User and Admin can visit the web page by this link to get more analysis details.

Going a step further, the tool's Sequence Diagram is as follow(doesn't contain admin):



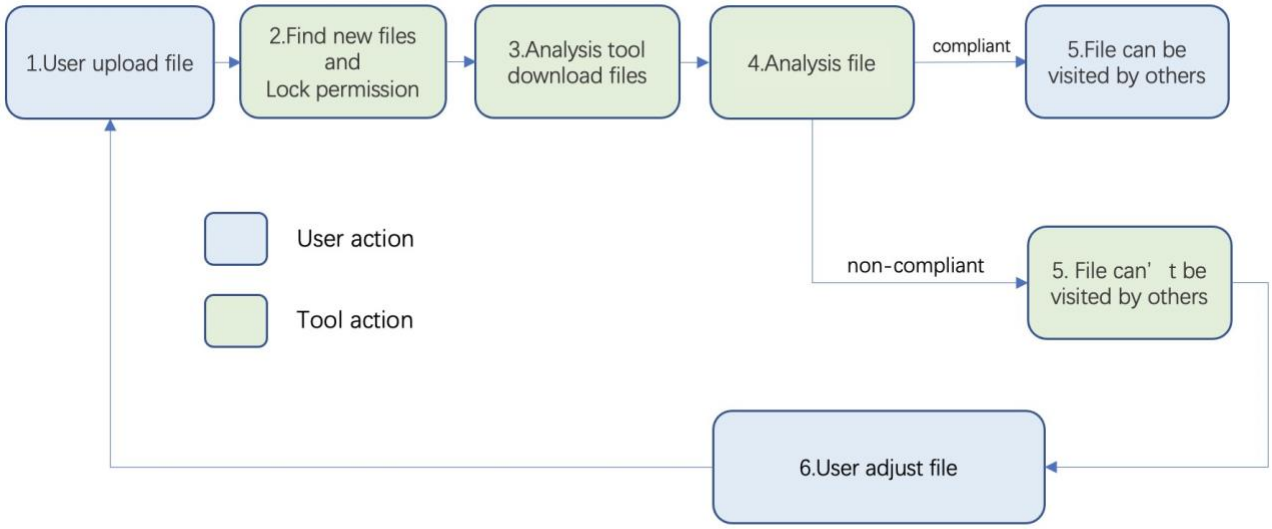
About new file monitor plugin in Antifactory:

When a file was uploaded to the Artifactory, file analysis tool need to find this file as soon as possible. Arifactory supports third-path plugins, these plugins can be triggered when certain behaviors occur in Artifactory(such as upload).

So design a file upload monitor plugin is a good choice. When the plugin finds a new file, it will send a message to the file analysis tool via network protocol. details

4.1.2. File upload and analysis flow

Analysis from another angle, this diagram depicts how the file flows between the each steps



1. User upload a file by Artifactory web application(or use some upload tools)
2. File analysis tool finds a new file was uploaded to the Artifactory and locks file's permission. Ensuring only file upload user,Artifactory Admin and file analysis tool can visit this file.
3. File analysis tool downloads the file.
4. Tool analyses the file context and determines whether it is compliant. Analysis strategy
5. If the file is compliant, tool will unlock the permission of the file, allow other users visit this file. If the file is non-compliant ,tool will keep permission locked and notify the user why file can't pass the analysis.
6. The file processing process has been completed. If file is non-compliant, user can adjust files by analysis result.

5. Appendix

5.1. file format identification

Tika website

https://tika.apache.org/

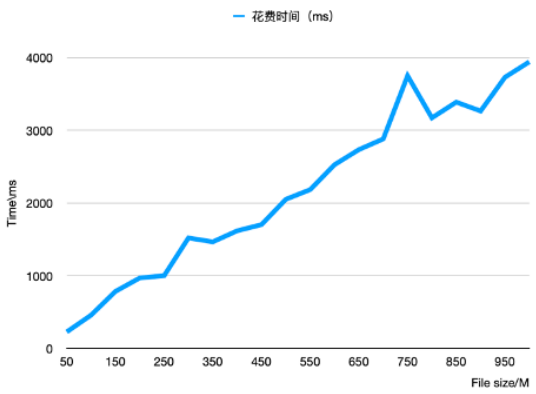
https://github.com/apache/tika

5.2. Apache Tika efficiency test

Tika program crashes when file size is 1050mb. So it is unstable when reading large files.

Tika文件读取效率测试

文件大小（M）	花费时间（ms）
50	230
100	461
150	787
200	972
250	1003
300	1522
350	1468
400	1620
450	1705
500	2053
550	2186
600	2529
650	2736
700	2883
750	3751
800	3173
850	3389
900	3267
950	3731
1000	3943
1050	程序崩溃



Tika 文件大小-读取时间 折线图

3.3. Time Estimation

Conservatively estimate, temporarily set the upper limit waiting time for file analysis at 1h

5.3. Federate

Federate Websitehttps://prod.ep.federate.a2z.com/
Federate SDKhttps://code.amazon.com/packages/Federate-oidc-oauth2-sdk/trees/mainline

Federate videohttps://broadcast.amazon.com/videos/339096

5.4. New file detection

5.4.1. Monitor Access-log

Artifactory will record every file operations in a log-file which is called Accesss-log. So Write a script to monitor Access-log. If monitor script find a new upload log in access-log file, it will send new file's information to file analysis tool by network protocol(the information include: file name,file path,file upload time,file user).

5.4.2. Artifactory plugin

Artifactory plugins are written by Groovy, which is essentially a script. These scripts can be triggered based on "Execution Points". Execution Points is an event such as upload and download.

jfrog websitehttps://www.jfrog.com/confluence/display/JFROG/User+Plugins

So file upload can trigger the plugin to run. After that the script can send the file information to file analysis tool.

	Access-log	Artifactory plugin

5.4.3. Comparison

efficiency	low Monitor script need to monitor changes in Access-log	high Plugins are trigger by "Execution Point". So it will run as soon as the file is uploaded
security	high Access-log is file.So monitor it won't affect the running of Artifactory	low If plugin runs with a error, The running of Artifactory will be affected

Tentative use a plugin to monitor for new file uploads.

5.5. Security policy

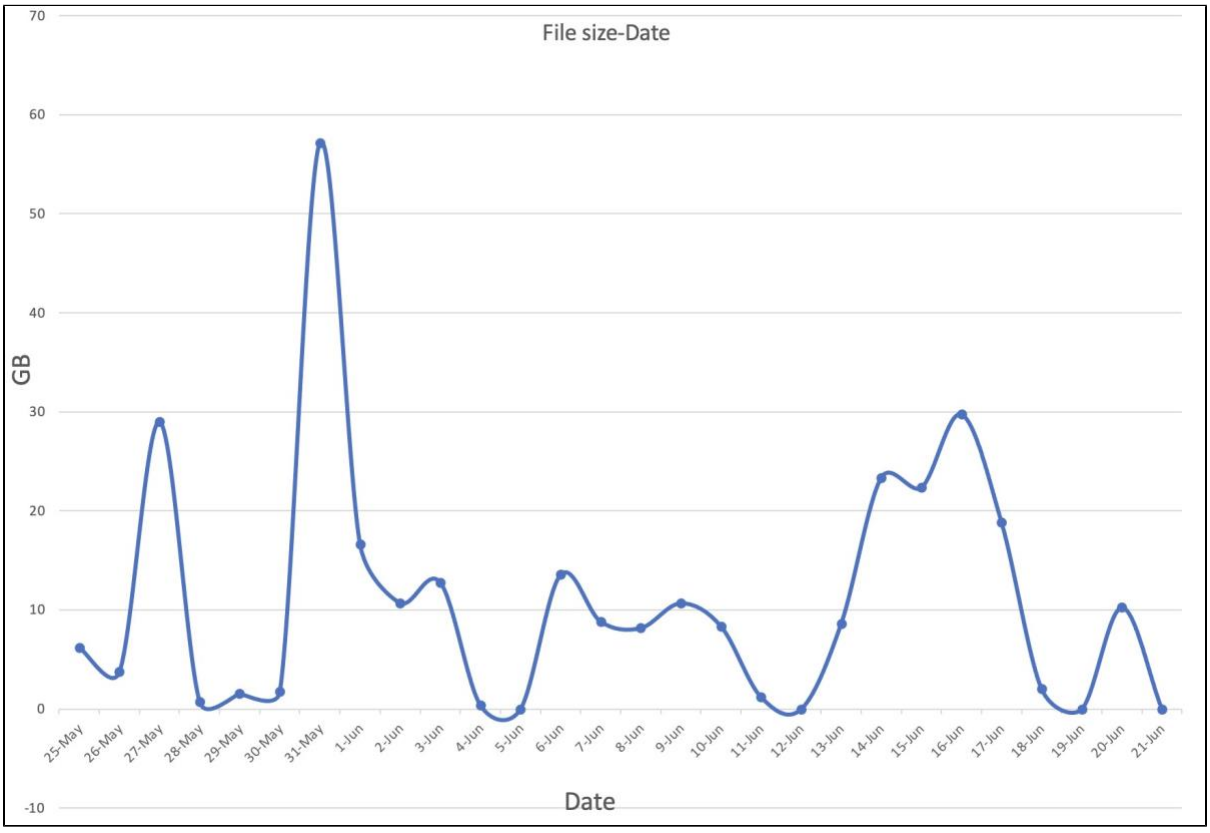
https://policy.amazon.com/policy/95
https://policy.amazon.com/policy/107

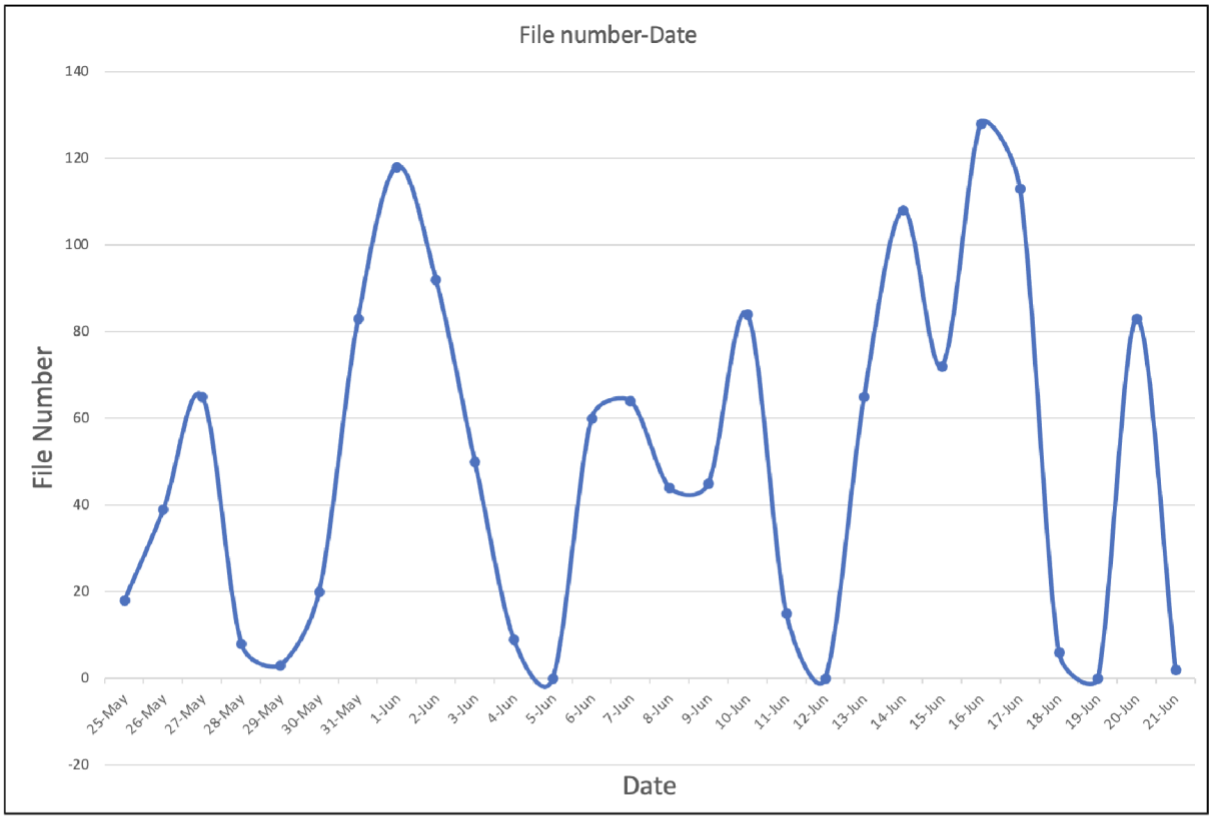
5.6. Artifactory file size analysis

By analyzing Artifactory's request log, line chart can be obtained. About request log: https://www.jfrog.com/confluence/display/JFROG/Logging

There are 10GB-13GB files will be uploaded to Artifactory each day on average.

The day with the most file uploads has 57g files were uploaded. During the weekend, the number of uploads dropped significantly.





Date	File size	File number
25-May	6.2098	18
26-May	3.7922	39
27-May	28.991	65
28-May	0.7219	8
29-May	1.5524	3
30-May	1.7727	20
31-May	57.1729	83
1-Jun	16.6044	118
2-Jun	10.7087	92
3-Jun	12.7804	50
4-Jun	0.3988	9
5-Jun	0	0
6-Jun	13.5634	60
7-Jun	8.8419	64
8-Jun	8.1856	44
9-Jun	10.6878	45
10-Jun	8.3638	84
11-Jun	1.2172	15
12-Jun	0	0
13-Jun	8.6408	65
14-Jun	23.378	108

15-Jun	22.3588	72
16-Jun	29.7494	128
17-Jun	18.8267	113
18-Jun	2.0262	6
19-Jun	0	0
20-Jun	10.2443	83
21-Jun	0.0027	2

5.7. Artifactory files data

I analyzed all files in the repo: "vendor_keira". The following is the analysis result report

There are 950 files in vendor_keira. The total file size is 185.25G.

The following table is the data of some files

	number	MIME type	detail	Pattern example
1	87761	text/x-python	py file	1.if __name__ =='__main__' 2.import
2	67168	application/octet-stream	Include dmgbinpyc	
3	56472	text/plain	Text files Include configuration files, log, gitignore, etc.	
4	44724	application/x-sh	shell secript	1. echo 2. printf 3. xxx(){} 4. ${}
5	27042	application/zlib	zlib files	
6	3524	multipart/appledouble	macOs config	
7	2957	application/x-sharedlib	adbfastboot	
8	2882	text/x-log	log About 9Gbit file	
9	2638	application/gzip	gzip	
10	2166	text/x-chdr	.h	1. #ifndef 2. #define 3. #endif
11	2118	application/json	json	
12	1537	application/zip	zip	
13	1438	application/javascript	js	
14	1209	text/html	html	<!DOCTYPE html> <html> <head> <title></title> </head> <body> </body> </html>
15	1090	application/x-matlab-data	matlab	
16	1075	text/csv	csv	
17	975	text/x-web-markdown	markdown	
18	632	application/xml	xml	