

YAML Configuration File

Overview

From version 5.8, Artifactory provides a flexible way to configure an existing Artifactory instance using a simple YAML configuration file. Run a single or multiple configuration changes as needed, to create, update and delete any elements in Artifactory.



Take care when modifying Artifactory configurations

Modifying the Artifactory configurations is an advanced feature, and if done incorrectly may render Artifactory in an undefined and unusable state. Since it is easy to overwrite configurations, such as a repository's package type, we strongly recommend backing up the configuration before making any direct changes, and taking great care when doing so.

YAML File Format

The YAML file is constructed with **keys** and **entities**, using the following format:

key: [entity]

For example:

```
localRepositories: #Local repositories configuration
  docker-local: #The local repository name
    type: docker #The package type
    ...
  ...
```

Page contents

- [Overview](#)
- [YAML File Format](#)
 - [Applying Configuration Changes](#)
 - [Deleting an Existing Element](#)
- [Supported Configurations](#)
 - [Repositories](#)
 - [Replication \(push and pull\)](#)
 - [General \(General, Property Sets, Proxy, Mail\)](#)
 - [Security \(General security, Password Policy, LDAP, SAML, OAuth, HTTP SSO, Crowd\)](#)
 - [Services configuration \(Backups, Maven Indexer\)](#)
 - [Advanced](#)

Applying Configuration Changes

Once you have configured your YAML file to include all the configuration changes needed, you can apply them by sending a PATCH request to `<host>:<port>/api/system/configuration`.

For example:

```
curl -u<username>:<password> -X PATCH "http://localhost:8081/artifactory/api/system/configuration" -H "Content-Type:application/yaml" -T configuration.yml
```



Admin privileges

You must supply a user with **Admin** privileges through the REST API.



Take care when modifying Artifactory configurations

Modifying the Artifactory configurations is an advanced feature, and if done incorrectly may render Artifactory in an undefined and unusable state. Since it is easy to overwrite configurations, such as a repository's package type, we strongly recommend backing up the configuration before making any direct changes, and taking great care when doing so.

Deleting an Existing Element

The YAML file can also be configured to delete elements, such as deleting a value for an existing field or removing an existing repository, using the following format: (e.g. some local repository) / deleting the value of an existing field (e.g. remove the "RnD maven repository" is as follows

key: null

Or

key: ~

For example:

Deleting a repository

```
localRepositories:  
  docker-local: null
```

```
localRepositories:  
  docker-local: ~
```

Deleting a field

```
remoteRepositories:  
  docker-remote:  
    description: null  
    notes: null  
    proxy: null
```

Supported Configurations

Repositories

```
localRepositories: #Local repositories configuration
  my-local-repo: #The local repository name
    type: maven | gradle | ivy | helm | sbt | nuget | gems | npm | bower | debian | composer | pypi |
docker | vagrant | gitlfs | yum | conan | chef | puppet | generic #The package type
    checksumPolicyType: server-generated-checksums | client-checksums #Determines how Artifactory
behaves when a client checksum for a deployed resource is missing or conflicts with the locally
calculated checksum
    description: "Hello world" #A free text field that describes the content and purpose of the
repository
    dockerApiVersion: V1 | V2 #Only for Docker repositories. The Docker API version
    excludesPattern: #Provides a way to filter out specific repositories when trying to resolve the
location of different artifacts (Ant-like pattern)
    includesPattern: #Provides a way to filter out specific repositories when trying to resolve the
location of different artifacts (Ant-like pattern)
    maxUniqueSnapshots: 0 #Only for Maven and NuGet repositories. Specifies the maximum number of
unique snapshots of the same artifact that should be stored. Once this number is reached and a new
snapshot is uploaded, the oldest stored snapshot is removed automatically. Default: 0
    maxUniqueTags: 0 #Only for Docker repositories. Specifies the maximum number of unique snapshots
of the same artifact that should be stored. Once this number is reached and a new snapshot is uploaded,
the oldest stored snapshot is removed automatically. Default: 0
    notes: "Hello world 2" #A free text field to add additional notes about the repository
    propertySets: #A list of property sets that will be defined for this repository
      - set1
      - set2
    repoLayout: maven-2-default #Sets the layout that the repository should use for storing and
identifying modules. The layout usually varies for the different package types
    snapshotVersionBehavior: unique | non-unique | deployer #Specifies the naming convention for Maven
SNAPSHOT versions
    xray: #Xray settings for the certain repository
      enabled: false #Indicated whether the repository will be indexed and scanned by Xray
    yumGroupFileNames: yumGroup1 #A comma-separated list of YUM group files associated with your RPM
packages
    yumRootDepth: 0 #Informs Artifactory under which level of directory to search for RPMs and save
the repodata directory
    blackedOut: false #If set, Artifactory ignores this repository when trying to resolve artifacts.
The repository is also not available for download or deployment of artifacts
    calculateYumMetadata: false #When set, RPM metadata calculation is automatically triggered by the
actions described above
    debianTrivialLayout: false #Indicates whether the Debian repository uses a trivial layout
    enableFileListsIndexing: false #When set, Artifactory will index and maintain an RPM filelists.xml
metadata file
    forceNugetAuthentication: false #When set, if anonymous access is enabled on the Artifactory
instance, this will override the behavior and will force client to authenticate
    handleReleases: true #If set, Artifactory allows you to deploy release artifacts into this
repository
    handleSnapshots: true #If set, Artifactory allows you to deploy snapshot artifacts into this
repository
    suppressPomConsistencyChecks: true #When deploying an artifact to a repository, Artifactory
verifies that the value set for groupId:artifactId:version in the POM is consistent with the deployed
path. If there is a conflict between these then Artifactory will reject the deployment.
    downloadRedirect: #Settings for redirecting download requests for direct download from cloud
storage
      enabled: false #When true, Artifactory will respond to download requests to this repository with
an HTTP 302 redirection return code along with a signed URL directing the calling client to download the
requested artifact directly from cloud storage.
```

```
remoteRepositories: #Remote repositories configuration
  my-remote-repo: #The remote repository name
    type: maven | gradle | ivy | helm | sbt | nuget | gems | npm | bower | debian | composer | pypi |
docker | vagrant | gitlfs | yum | conan | chef | puppet | generic #The package type
    url: https://jcenter.bintray.com #The URL for the remote repository. Needs to be only HTTP and
HTTPS URLs
    username: user #The remote repository username, when authentication is required
    password: password #The remote repository password, when authentication is required

    assumedOfflinePeriodSecs: 300 #The number of seconds the repository stays in assumed offline state
after a connection error. At the end of this time, an online check is attempted in order to reset the
offline status. A value of 0 means the repository is never assumed offline.
```

remoteRepoChecksumPolicyType: generate-if-absent | fail | ignore-and-generate | pass-thru #Only for Maven repositories. The Checksum Policy determines how Artifactory behaves when a client checksum for a remote resource is missing or conflicts with the locally calculated checksum

clientTlsCertificate: tls-certificate #The client TLS certificate (alias) that will be used when authenticating against the remote resource.

description: "Hello world" #A free text field that describes the content and purpose of the repository

dockerApiVersion: V1 | V2 #Only for Docker repositories. The Docker API version

excludesPattern: excludingremotel #Provides a way to filter out specific repositories when trying to resolve the location of different artifacts (Ant-like pattern)

includesPattern: includingremotel #Provides a way to filter out specific repositories when trying to resolve the location of different artifacts (Ant-like pattern)

maxUniqueSnapshots: 0 #Only for Maven and VCS repositories. Specifies the maximum number of unique snapshots of the same artifact that should be stored. Once this number is reached and a new snapshot is uploaded, the oldest stored snapshot is removed automatically. Default: 0

maxUniqueTags: 0 #Only for Docker repositories. Specifies the maximum number of unique snapshots of the same artifact that should be stored. Once this number is reached and a new snapshot is uploaded, the oldest stored snapshot is removed automatically. Default: 0

blockMismatchingMimeTypes: true #If set, artifacts will fail to download if a mismatch is detected between requested and received mimetype, according to the list specified in the system properties file under blockedMismatchingMimeTypes. You can override by adding mimetypes to the override list in the mismatchingMimeTypesOverrideList field

mismatchingMimeTypesOverrideList: xml #When set, artifacts will fail to download if a mismatch is detected between the requested and received mime type, according to a list specified in the system properties file under blockedMismatchingMimeTypes. You can override this setting by adding mime types to this field

missedRetrievalCachePeriodSecs: 1800 #If a remote repository is missing a requested artifact, Artifactory will return a "404 Not found" error. This response is cached for the period of time specified by this parameter. During that time, Artifactory will not issue new requests for the same artifact. A value of 0 means that the response is not cached and Artifactory will always issue a new request when demanded

notes: "Hello world 2" #A free text field to add additional notes about the repository

propertySets: #A list of property sets that will be defined for this repository

- set1
- set2

proxy: proxy1 #The proxy name configured in the Proxy section to use for accessing the remote resource

queryParams: query-params&for-my-remote-repo #Custom HTTP query parameters that will be automatically included in all remote resource requests. For example: param1=val1¶m2=val2¶m3=val3

repoLayout: nameremotel #Sets the layout that the repository should use for storing and identifying modules

retrievalCachePeriodSecs: 600 #This value refers to the number of seconds to cache metadata files before checking for newer versions on remote server. A value of 0 indicates no caching

socketTimeoutMillis: 15000 #Network timeout (in ms) to use when establishing a connection and for unanswered requests. Timing out on a network operation is considered a retrieval failure

unusedArtifactsCleanupPeriodHours: 0 #The number of hours to wait before an artifact is deemed "unused" and eligible for cleanup from the repository. A value of 0 means automatic cleanup of cached artifacts is disabled

allowAnyHostAuth: false #Allow credentials of this repository to be used on requests redirected to any other host

blackedOut: false #When set, the repository or its local cache do not participate in artifact resolution

bypassHeadRequests: false #Before caching an artifact, Artifactory first sends a HEAD request to the remote resource. In some remote resources, HEAD requests are disallowed and therefore rejected, even though downloading the artifact is allowed. When set, Artifactory will bypass the HEAD request and cache the artifact directly using a GET request

enableCookieManagement: false #Enables cookie management if the remote repository uses cookies to manage client state

fetchJarsEagerly: false #Only for Maven repositories. When set, the repository attempts to eagerly fetch the jar in the background each time a POM is requested

fetchSourcesEagerly: false #When set, the repository attempts to eagerly fetch the source jar in the background each time a jar is requested

forceNuggetAuthentication: false #Force basic authentication credentials in order to use this repository

handleReleases: true #If set, Artifactory allows you to deploy release artifacts into this repository

handleSnapshots: true #If set, Artifactory allows you to deploy snapshot artifacts into this repository

listRemoteFolderItems: true #Lists the items of remote folders in simple and list browsing. Required for dynamic resolution that depends on remote folder content information, such as remote Ivy version lookups. The remote content is cached according to the value of the 'Retrieval Cache Period'

```
offline: false #If set, Artifactory does not try to fetch remote artifacts. Only locally-cached
artifacts are retrieved
synchronizeProperties: false #When set, remote artifacts are fetched along with their properties

xray: #Xray settings for the certain repository
  enabled: false #Indicated whether the repository will be indexed and scanned by Xray

bower: #Only for Bower repositories
  bowerRegistryUrl: http://someurl.com #The registry URL from which packages will be resolved

cocoaPods: #Only for CocoaPods repositories
  cocoaPodsSpecsRepoUrl: http://someurl.com #The specs URL. When using a private specs repo, set
the URL to be the same as the one configured in the URL field
  specRepoProvider: #The specs repo provider settings
    provider: github | bitbucket | stash | oldstash | artifactory | custom #The Git Provider the
remote repository works against
    downloadUrl: http://someurl.com #The custom download URL. Need to set only if the provider is
set to 'custom'

nuget: #Only for NuGet repositories
  downloadContextPath: api/v2/package #The context path prefix through which NuGet downloads are
served. For example, the NuGet Gallery download URL is 'https://nuget.org/api/v2/package', so the
repository URL should be configured as 'https://nuget.org' and the download context path should be
configured as 'api/v2/package'
  feedContextPath: api/v2 #The context path prefix through which the NuGet feeds are served. For
example, the NuGet Gallery feed URL is 'https://nuget.org/api/v2', so the repository URL should be
configured as 'https://nuget.org' and the feed context path should be configured as 'api/v2'

composer: #Only for PHP Composer repositories
  composerRegistryUrl: http://someurl.com #The registry URL from which packages will be resolved

vcs: #Only for VCS, CocoaPods, Bower and Composer repositories
  git:
    provider: github | bitbucket | stash | oldstash | artifactory | custom #The Git Provider the
remote repository works against
    downloadUrl: http://someurl.com #The custom download URL. Need to set only if the provider is
set as CUSTOM
    type: git #Mandatory only when creating a VCS remote repository
    downloadRedirect: #Settings for redirecting download requests for direct download from cloud
storage
    enabled: false #When true, Artifactory will respond to download requests to this repository with
an HTTP 302 redirection return code along with a signed URL directing the calling client to download the
requested artifact directly from cloud storage.
```

```

virtualRepositories: #Virtual repositories configuration
  virtual1: #The virtual repository name
    type: maven | gradle | ivy | helm | sbt | nuget | gems | npm | bower | debian | composer | pypi |
docker | vagrant | gitlfs | yum | conan | chef | puppet | generic #The package type
    repositories: #A list of local and remote repositories that will be aggregated under this virtual
repository
      - my-local-repo
      - my-remote-repo
    description: "" #A free text field that describes the content and purpose of the repository
    notes: "" #A free text field to add additional notes about the repository
    dockerApiVersion: V1 | V2 #Only for Docker repositories. The Docker API version
    includesPattern: #Provides a way to filter out specific repositories when trying to resolve the
location of different artifacts (Ant-like pattern)
    excludesPattern: #Provides a way to filter out specific repositories when trying to resolve the
location of different artifacts (Ant-like pattern)
    externalDependencies: #Only for npm repositories. External dependencies configuration.
      enabled: false #When set, dependency rewrite for the specified npm repository will be enabled
      patterns: #Optional include patterns to match external dependencies. Ant-style path expressions
are supported (*, **, ?). For example, specifying **/github.com/** will only allow external dependencies
from github.com host
        - "**/github.com/**"

    keyPair: keypairvirtual1 #Only for Maven repositories, for WebStart and Jar signing. The alias of
the keypair configured in the Signing Keys page
    p2: #Only for P2 repositories
      urls: #A list of remote URLs that are aggregate under this virtual repositorys
        - http://someurl.com
        - http://someurl2.com
    pomRepositoryReferencesCleanupPolicy: discard_any_reference | discard_active_reference | nothing
#Only for Maven repositories. (1) Discard Active References - Removes repository elements that are
declared directly under project or under a profile in the same POM that is activeByDefault. (2) Discard
Any References - Removes all repository elements regardless of whether they are included in an active
profile or not. (3) Nothing - Does not remove any repository elements declared in the POM.
    repoLayout: maven-2-default #Sets the layout that the repository should use for storing and
identifying modules
    virtualCacheConfig:
      virtualRetrievalCachePeriodSecs: 600 #This value refers to the number of seconds to cache
metadata files before checking for newer versions on aggregated repositories. A value of 0 indicates no
caching
      artifactoryRequestsCanRetrieveRemoteArtifacts: false #Determines whether artifact requests coming
from other instance of Artifactory can be fulfilled by accessing this virtual repository's remote
repositories, or by only accessing its caches (default)
      forceNugetAuthentication: false #Only for NuGet repositories. Force basic authentication
credentials in order to use this repository

```

Replication (push and pull)

```
localReplications: #Replication configuration for local repositories (push replication - while the
local repository is the source)
  [my-replicationKey]: #A unique key to identify this replication setting. Replace [my-replicationKey]
with some string. This string will be needed when performing changes on this specific replication setting
    repoKey: my-local-repo #Local repository name to replicate from
    cronExp: "0 0 12 * * ?" #The Cron expression that determines the frequency of replication
    url: http://someurl.com:8081/artifactory/repo #The URL of the target local repository on a remote
Artifactory server
  username: user #The HTTP authentication username
  password: password #The HTTP authentication username
  pathPrefix: /org/my-company/jars #Only artifacts that located in path that matches the subpath
within the repository will be replicated
  proxy: proxy1 #A proxy configuration to use when communicating with the remote instance
  socketTimeoutMillis: 15000 #The network timeout in milliseconds to use for remote operations
  enableEventReplication: true #When set, each event will trigger replication of the artifacts
changed in this event. This can be any type of event on artifact, e.g. added, deleted or property change
  enabled: false
  syncDeletes: false #When set, items that were deleted locally should also be deleted remotely
(also applies to properties metadata)
  syncProperties: false #When set, the task also synchronizes the properties of replicated artifacts
  syncStatistics: false #When set, the task also synchronizes artifact download statistics. Set to
avoid inadvertent cleanup at the target instance when setting up replication for disaster recovery
```

```
remoteReplications: #Replication configuration for remote repositories (pull replication - while the
rremote repository is the source)
  [my-replicationKey]: #A unique key to identify this remote replication setting. Replace [my-
replicationKey] with some string. This string will be needed when performing changes on this specific
replication setting
    repoKey: my-remote-repo #Remote repository name to replicate from
    cronExp: "0 0 12 * * ?" #The Cron expression that determines the frequency of replication
    pathPrefix: pathPrefixlocal1 #Only artifacts that located in path that matches the subpath within
the repository will be replicated
    enableEventReplication: false #When set, each event will trigger replication of the artifacts
changed in this event. This can be any type of event on artifact, e.g. added, deleted or property change
    enabled: false
    syncDeletes: false #When set, items that were deleted locally should also be deleted remotely
(also applies to properties metadata)
    syncProperties: true #When set, the task also synchronizes the properties of replicated artifacts
```

General (General, Property Sets, Proxy, Mail)

```
urlBase: https://someurl:8081/artifactory/ #A custom URL base is useful when Artifactory is running
behind a proxy. In this case the base for URLs generated in Artifactory for links and redirect responses
must be specified manually
fileUploadMaxSizeMb: 100 #The maximum number artifacts that may be downloaded under one folder
dateFormat: dd-MM-yy HH:mm:ss z #The date format for displaying dates in the web interface
offlineMode: false #If set, Artifactory does not try to access remote resources to fetch artifacts.
Only cached and local artifacts are served
```

```
folderDownloadConfig: #Folder download configuration
  enabled: false #When set, folder download will be enabled
  maxConcurrentRequests: 10 #The maximum number of folder download requests that may be run
concurrently
  maxDownloadSizeMb: 1024 #The maximum size (in MB) of a folder that may be downloaded
  maxFiles: 5000 #The maximum number artifacts that may be downloaded under one folder
  enabledForAnonymous: false #If enabled is set to true, setting this to true will allow anonymous
users to download folders
```

```
replicationsConfig: #Global replication config
  blockPullReplications: false #When set, pull replication will be blocked regardless of the
configuration for specific repositories
  blockPushReplications: false #When set, push replication will be blocked regardless of the
configuration for specific repositories
```

```
systemMessageConfig: #System message configuration
  enabled: false #When set, the system message will be displayed
  message: "Some message" #The message to display. You can include links in the message using the
following format: [<link URL>, <link text>]
  title: "Some title" #The title for the message
  titleColor: color #The color in Hex format (e.g. 99c27c)
  showOnAllPages: false #When set, the custom message will be displayed on all pages in the UI. When
not set, the custom message will only be displayed on the Artifactory Home Page
```

```
trashcanConfig: #Trash can configuration
  enabled: true #If set, trash can will be enabled and deleted items will be stored in the trash can
for the specified retention period
  retentionPeriodDays: 14 #The number of days to keep deleted items in the trash can before deleting
permanently
```

```
propertySets: #Property sets configuration
  property-set1: #Property set name
  properties: #A list of properties that will be part of the property set
    - name: "prop1"
      predefinedValues:
        passed-QA:
          defaultValue: false
        dev-QA:
          defaultValue: false
      closedPredefinedValues: false
      multipleChoice: false
    - name: "prop2"
      predefinedValues:
        passed-QA:
          defaultValue: false
        dev-QA:
          defaultValue: false
      closedPredefinedValues: false
      multipleChoice: false
  visible: true
```

```
proxies: #Proxies configuration
  proxy1: #The unique ID of the proxy setting
  domain: domainproxy1 #The proxy domain/realm name
  host: hostproxy1 #The name of the proxy host
  ntHost: nthostproxy1 #The computer name of the machine (the machine connecting to the NTLM proxy)
  password: password #The proxy password when authentication credentials are required
  port: 0 #The proxy port number
  redirectedToHosts: redirectedproxy1 #An optional list of newline or comma separated host names to
which this proxy may redirect requests. The credentials defined for the proxy are reused by requests
redirected to all of these hosts.
  username: user #The proxy username when authentication credentials are required
  defaultProxy: false #When set, this proxy will be the default proxy for new remote repositories
and for internal HTTP requests issued by Artifactory
```

```
mailServer: #Mail server settings
  enabled: true #When set, mail notifications are enabled
  artifactoryUrl: http://someurl.com #The Artifactory URL to to link to in all outgoing messages
(optional)
  from: john@company.com #The "from" address header to use in all outgoing messages (optional)
  host: 10.1.1.1 #The mail server IP address / DNS
  username: user #The username for authentication with the mail server
  password: password #The password for authentication with the mail server
  port: 25 #The port number of the mail server
  subjectPrefix: "Notice" #A prefix to use for the subject of all outgoing mails
  ssl: false #When set, uses a secure connection to the mail server
  tls: false #When set, uses Transport Layer Security when connecting to the mail server
```

Security (General security, Password Policy, LDAP, SAML, OAuth, HTTP SSO, Crowd)

```
security: #Security configuration (LDAP, SAML, Password Policy, ...)
  anonAccessEnabled: true #When set, anonymous access will be enabled for the set of permissions
  assigned to the default "anonymous user"
  anonAccessToBuildInfosDisabled: false #Deprecated from Artifactory version 6.6 #This setting gives
  you more control over anonymous access, and allows you to prevent anonymous users from accessing the
  Build module where all information related to builds is found, even when anonymous access is enabled.
```

```
security: #Security configuration (LDAP, SAML, Password Policy, ...)
  userLockPolicy: #User lock policy configuration
  enabled: false #When set, the lock policy will be enabled
  loginAttempts: 5 #Lock user after exceeding max failed login attempts
```

```
security: #Security configuration (LDAP, SAML, Password Policy, ...)
  passwordSettings: #Password settings
  encryptionPolicy: REQUIRED | SUPPORTED | UNSUPPORTED #Determines the password requirements from
  users identified to Artifactory from a remote client such as Maven. The options are: (1) Supported
  (default): Users can authenticate using secure encrypted passwords or clear-text passwords. (2)
  Required: Users must authenticate using secure encrypted passwords. Clear-text authentication fails. (3)
  Unsupported: Only clear-text passwords can be used for authentication
  expirationPolicy: #Password expiration policy
  enabled: false #When checked, password expiration policy is enabled
  passwordMaxAge: 60 #The time interval in which users will be obligated to change their password
  notifyByEmail: true #When set, users receive an email notification a few days before their
  password expires
```

```
security: #Security configuration (LDAP, SAML, Password Policy, ...)
  ldapSettings: #LDAP server(s) settings
    ldap1: #The unique ID of the LDAP setting
      emailAttribute: email1 #An attribute that can be used to map a user's email to a user created
      automatically by Artifactory
      ldapPoisoningProtection: true #When set to true (recommended), Artifactory will protect against
      LDAP poisoning by filtering out users exposed to vulnerability
      ldapUrl: ldap://myserver:myport/dc=sampledomain,dc=com #Location of the LDAP server in the
      following format: ldap://myserver:myport/dc=sampledomain,dc=com. The URL should include the base DN used
      to search for and/or authenticate users
      search:
        managerDn: manager1 #The full DN of a user with permissions that allow querying the LDAP
        server. When working with LDAP Groups, the user should have permissions for any extra group attributes
        such as memberOf
        managerPassword: managerpass1 #The password of the user binding to the LDAP server when using
        "search" authentication
        searchBase: searchbase1 #The Context name in which to search relative to the base DN in the
        LDAP URL. Multiple search bases may be specified separated by a pipe ( | ). This is parameter is optional
        searchFilter: searchfilter1 #A filter expression used to search for the user DN that is used
        in LDAP authentication. Possible examples are: uid={0}) - this would search for a username match on the
        uid attribute. Authentication using LDAP is performed from the DN found if successful
        searchSubTree: true #When set, enables deep search through the sub-tree of the LDAP URL +
        Search Base
        userDnPattern: userppatt1 #A DN pattern used to log users directly in to the LDAP database. This
        pattern is used to create a DN string for "direct" user authentication, and is relative to the base DN
        in the LDAP URL. For example: uid={0},ou=People
        allowUserToAccessProfile: false #When set, users created after logging in using LDAP will be
        able to access their profile page in Artifactory
        autoCreateUser: true #When set, Artifactory will automatically create new users for those who
        have logged in using LDAP, and assign them to the default groups
        enabled: true #When set, these settings are enabled
```

```
security: #Security configuration (LDAP, SAML, Password Policy, ...)
  ldapGroupSettings: #LDAP group(s) settings
    name1: #The unique ID of the LDAP group setting
      descriptionAttribute: desc1 #An attribute on the group entry which denoting the group
      description. Used when importing groups
      enabledLdap: enabled1 #The LDAP setting (from the ldapSettings section) you want to use for
      group retrieval
      filter: filter1 #The LDAP filter used to search for group entries. Used when importing groups
      groupBaseDn: groupbase1 #A search base for group entry DNs, relative to the DN on the LDAP
      server's URL (and not relative to the LDAP Setting's "searchBase"). Used when importing groups
      groupMemberAttribute: uniqueMember #A multi-value attribute on the group entry containing user
      DNs or IDs of the group members (e.g., uniqueMember,member)
      groupNameAttribute: groupName #Attribute on the group entry denoting the group name. Used when
      importing groups
      strategy: STATIC | DYNAMIC | HIERARCHY #Group synchronization strategy
      subTree: false
```

```
security: #Security configuration (LDAP, SAML, Password Policy, ...)
  crowdSettings: #Crowd / JIRA users management configuration
    applicationName: myApp #The application name configured for Artifactory in Crowd / JIRA
    password: password #The application password configured for Artifactory in Crowd / JIRA
    serverUrl: http://someurl.com #The full URL of the server to use
    sessionValidationInterval: 0 #The time window, in minutes, in which the session does not need to be
    revalidated
    enableIntegration: false #Set this checkbox to enable security integration with Atlassian Crowd or
    JIRA
    noAutoUserCreation: false #When set to true, authenticated users will not be automatically created
    inside Artifactory. Instead, for every request from a Crowd / JIRA user, the user is temporarily
    associated with default groups (if such groups are defined), and the permissions for these groups applies
    useDefaultProxy: false #If set and a default proxy definition exists, it is used to pass through
    to the Crowd / JIRA Server
```

```
security: #Security configuration (LDAP, SAML, Password Policy, ...)
  samlSettings: #SAML SSO settings
    enableIntegration: false #When set, SAML integration is enabled and users may be authenticated via
a SAML server
    certificate: certificate #The X.509 certificate that contains the public key. The certificate must
contain the public key to allow Artifactory to verify sign-in requests
    emailAttribute: #If noAutoUserCreation is set to false or an internal user exists, Artifactory
will set the user's email to the value in this attribute that is returned by the SAML login XML response.
    groupAttribute: #The group attribute in the SAML login XML response
    loginUrl: http://someurl.com/login #The identity provider login URL (when you try to login, the
service provider redirects to this URL)
    logoutUrl: http://someurl.com/logout #The identity provider logout URL (when you try to logout, the
service provider redirects to this URL)
    noAutoUserCreation: true #When set, for every request from a SAML user, the user is temporarily
associated with default groups (if such groups are defined), and the permissions for these groups apply.
Without automatic user creation, you must manually create the user inside Artifactory to manage user
permissions not attached to their default groups. When not set, authenticated users are automatically
created in Artifactory.
    serviceProviderName: serviceProvider #The Artifactory name in the SAML federation
    allowUserToAccessProfile: false #Auto created users will have access to their profile page and
will be able to perform actions such as generate API key
    autoRedirect: false #When set, clicking on the login link will direct users to the configured SAML
login URL
    syncGroups: false #When set, in addition to the groups the user is already associated with, he
will also be associated with the groups returned in the SAML login response. Note that the user's
association with the returned groups is not persistent. It is only valid for the current login session.
```

```
security: #Security configuration (LDAP, SAML, Password Policy, ...)
  oauthSettings: #OAuth SSO settings
    allowUserToAccessProfile: false #When set, users created after authenticating using OAuth, will be
able to access their profile. This means they are able to generate their API Key and set their password
for future use
    enableIntegration: true #If set to true, authentication with an OAuth provider is enabled and
Artifactory will display all OAuth providers configured. If not set, authentication is by Artifactory
user/password
    persistUsers: false #When set, Artifactory will create an Artifactory user account for any new
user logging in to Artifactory for the first time
    oauthProvidersSettings: #OAuth SSO provider (e.g. GitHub, Google) settings
      github-oauth: #The Unique ID of the OAuth Provider setting
        apiUrl: "https://api.github.com/user" #The URL used for API access, if needed to get user data
(e.g. https://api.github.com/user)
        authUrl: "https://github.com/login/oauth/authorize" #The URL used for the initial
authentication step (e.g. https://github.com/login/oauth/authorize)
        basicUrl: "https://github.com/" #The URL used to acquire a token via basic auth (e.g.
https://github.com/)
        enabled: false #When set, the OAuth SSO provider setting is enabled
        providerType: github | cloudfoundry | google | openid #The provider type
        secret: secret #The OAuth2 shared secret, given by the provider
        tokenUrl: "http://someurl.com/token" #The URL used to acquire a token from the provider
```

```
security: #Security configuration (LDAP, SAML, Password Policy, ...)
  httpSsoSettings: #HTTP SSO configuration
    httpSsoProxied: false #When set, Artifactory trusts incoming requests and reuses the remote user
originally set on the request by the SSO of the HTTP server
    remoteUserRequestVariable: remoter #The name of the HTTP request variable to use for extracting
the user identity. Default is: REMOTE_USER
    allowUserToAccessProfile: false #When set, users created after authenticating using HTTP SSO, will
be able to access their profile. This means they are able to generate their API Key and set their
password for future use
    noAutoUserCreation: false #When set to true, authenticated users will not be automatically created
inside Artifactory. Instead, for every request from a Crowd / JIRA user, the user is temporarily
associated with default groups (if such groups are defined), and the permissions for these groups applies
```

Services configuration (Backups, Maven Indexer)

```
backups: #Manage the automatic and periodic backups of the entire Artifactory instance
  backup-daily: #The backup name
    cronExp: "0 0 12 * * ? " #A Cron expression to control the backup frequency
    excludedRepositories: #A list of excluded repositories from the backup
      - local1
      - local2
    retentionPeriodHours: 1000 #The number of hours to keep a backup before Artifactory will clean it
up to free up disk space. Applicable only to non-incremental backups
    createArchive: false #If set, backups will be created within a Zip archive (Slow and CPU intensive)
    enabled: true
    excludeBuilds: false #(Relevant only from Artifactory 6.6 and below) Exclude all builds from the
backup
    excludeNewRepositories: false #When set, new repositories will not be automatically added to the
backup
    sendMailOnError: true #If set, all Artifactory administrators will be notified by email if any
problem is encountered during backup
```

```
indexer: #Maven indexer settings
  enabled: false #When set, the Maven indexer will be enabled and run according to the cronExp
  cronExp: abcder #A Cron expression that determines the frequency in which Maven indexes on the
selected repositories will be recalculated
  includedRepositories: #Specifies the repositories that should be indexed on the next run
    - local1
    - local2
```

Advanced

```
gcConfig: #Garbage collection settings
  cronExp: "0 0 12 * * ?" #The Cron expression that determines the frequency of garbage
collection.

quotaConfig: #Storage quota configuration
  diskSpaceLimitPercentage: 0 #The percentage of available disk space that may be used for storage
before Artifactory rejects deployments and issues errors
  diskSpaceWarningPercentage: 0 #The percentage of available disk space that may be used for storage
before Artifactory issues warnings
  enabled: false #When set, Artifactory will monitor disk space usage and issue warnings and errors
according to the quotas specified in diskSpaceLimitPercentage and diskSpaceWarningPercentage

cleanupConfig: #Cleanup Unused Cached Artifacts
  cronExp: "0 0 12 * * ?" #A Cron expression that determines the frequency of artifacts cleanup

virtualCacheCleanupConfig: #Virtual repositories cleanup configuration. The cleanup will delete cached
POM files
  cronExp: "0 0 12 * * ?" #A Cron expression to control the frequency of when to run the cleanup
```