17.0 Audit Log APIs

WAS THIS PAGE HELPFUL? Leave Feedback

17.0 Audit Logs APIs

17.1 Query Audit Logs

Retrieve audit log records by query. Audit logs exist primarily to provide reporting and are read-only in nature. As a result, audit logs are read by query and not individually. Multiple audit logs, returned most recent first, are returned wrapped in an XML `<auditLogs>` element or JSON object with an `auditLogs` array member. Since the number of audit logs is unbounded, paging parameters are essentially required to avoid excessive retrieval.

17.1.1 Method: GET Audit Logs

GET /api/auditlogs?query=(query criteria see below)

17.1.2 Method: GET Audit Logs by Host

GET /api/auditlogs/{hostName}

17.1.3 Method: GET Audit Logs by Host and Service

GET /api/auditlogs/{hostName}/{serviceDescription}

17.1.4 HTTP Query and Path Parameters

Field	Type	Description	Required
query	Query	An encoded query string (where clause)	no
first	Query	Paging, first record to start from	no
count	Query	Paging, the number of records to include when paging	no
hostName	Path	the name of the host to retrieve records for	no
serviceDescription	Path	the service description to retrieve records for	no

If neither hostName and serviceDescription path parameters nor query query parameter are provided, all audit log records will be retrieved. The first and count paging parameters are required to avoid excessive retrieval.

17.1.5 HTTP Headers

Header	Valid Values	Required
Accept	application/xml or application/json	False
GWOS-API-TOKEN	a valid token returned from login	True
GWOS-APP-NAME	your application name	True

17.1.6 Query Fields

The table below contains valid fields in the value of the 'query' query parameter.

Field	Description	Alias
auditLogId	audit log id	id
subsystem	subsystem audit action is logged for	
action	audit action logged, (`ADD`, `DELETE`, `MODIFY`, `SYNC`, `ENABLE`, `DISABLE`, `BACKUP`, `RESTORE`, `ACTION`)	

description	description of audit action logged	
username	name of user performing audited action	user
timestamp	time audit action logged	
hostName	name of host audit action is logged for	host
serviceDescription	unique description of service audit action is logged for	service
hostGroupName	name of host group audit action is logged for	hostGroup
serviceGroupName	name of service group audit action is logged for	hostGroup

Note: query fields are case-insensitive, thus camelCase, or all lower case will both work fine.

17.1.7 Example Queries

These examples are not HTTP encoded for readability. In practice queries must be encoded.

1. *query for all most recent audit logs, (most recent ordering returned by default)*
   GET /api/auditlogs?count=25
2. *query for most recent audit logs for a host, (most recent ordering returned by default)*
   GET /api/auditlogs/server_2?count=25
3. *query for most recent audit logs for specific service on a host, (most recent ordering returned by default)*
   GET /api/auditlogs/server_2/local_processes?count=25
4. *query for the second page of most recent audit logs for a user on a host, (most recent ordering must be specified explicitly when 'query' used)*
   GET /api/auditlogs?query=userName = 'admin' AND hostName = 'server_2' ORDER BY timestamp DESC, auditLogId DESC&first=25&count=25

17.1.8 HTTP Status Codes

Code	Description
200	Query returned no audit logs
200	Query returned one or more audit logs
401	Authentication/authorization error occurred
500	An internal server error occurred while querying audit logs

17.1.9 Example Query Results

Here is an XML example of the result of a query finding audit logs.

XML query results are always wrapped in an `<auditLogs>` collection element, with one or more `<auditLog>` subelements.

```
<auditLogs>
    <auditLog auditLogId="30" subsystem="SV" hostName="server_2" action="DELETE" description="Test
deletion server 2." username="admin" timestamp="2014-09-26T14:03:24.515-0600" serviceDescription=
"network_users"/>
    <auditLog auditLogId="29" subsystem="SV" hostName="server_2" action="ADD" description="Test
addition server 2." username="admin" timestamp="2014-09-26T14:03:24.515-0600" serviceDescription=
"local_processes"/>
    <auditLog auditLogId="28" subsystem="Console" hostName="server_1" action="MODIFY" description=
"Test modification server 1." username="admin" timestamp="2014-09-26T14:03:23.935-0600"/>
</auditLogs>
```

Here is a JSON example of the result of a query finding audit logs.

JSON query results are always wrapped in an object with a `auditLogs` array member, with one or more object members.

```
{
  "auditLogs": [ {
    "auditLogId" : 33,
    "subsystem" : "SV",
    "hostName" : "server_2",
    "action" : "DELETE",
    "description" : "Test deletion server 2.",
    "username" : "admin",
    "timestamp" : "2014-09-26T14:03:25.600-0600",
    "serviceDescription" : "network_users"
  }, {
    "auditLogId" : 32,
    "subsystem" : "SV",
    "hostName" : "server_2",
    "action" : "ADD",
    "description" : "Test addition server 2.",
    "username" : "admin",
    "timestamp" : "2014-09-26T14:03:25.600-0600",
    "serviceDescription" : "local_processes"
  }, {
    "auditLogId" : 31,
    "subsystem" : "Console",
    "hostName" : "server_1",
    "action" : "MODIFY",
    "description" : "Test modification server 1.",
    "username" : "admin",
    "timestamp" : "2014-09-26T14:03:25.033-0600"
  } ]
}
```


17.2 Create Audit Logs

Add audit logs by post. Any number of audit logs can posted to the server in a single asynchronous, (the default), or synchronous request. Timestamps and audit log ids will be overwritten by the server, so these fields should not be specified as part of the post body request. As noted above, the action data member is an enumerated type and is limited to these values: `ADD`, `DELETE`, `MODIFY`, `SYNC`, `ENABLE`, `DISABLE`, `BACKUP`, `RESTORE`, or `ACTION`. The remaining data members are stored as opaque strings without an sort of validation applied.

17.2.1 Method: POST Audit Logs

POST /api/auditlogs

17.2.2 Query Parameters

Field	Description	Required
async	A boolean flag to indicate whether to submit the batch of audit log inserts asynchronously or not. Default is true. Valid values are true or false. A synchronous request will not return back to your client code until the completion of processing all objects provided in the post data. Whereas an asynchronous request will submit the work to a queue, and return immediately.	False

17.2.3 HTTP Headers

Header	Valid Values	Required
Content-Type	application/xml or application/json	True
Accept	application/xml or application/json	True
GWOS-API-TOKEN	a valid token returned from login	True
GWOS-APP-NAME	your application name	True

17.2.4 POST Data Example

Here is an XML example post data creating two audit logs on the server, (the wrapping `<auditlogs>` collection element is always required, even

for a single audit log element):

```
<auditLogs>
    <auditLog subsystem="SV" hostName="server_2" action="ADD" description="Test addition server 2."
username="admin" serviceDescription="local_processes"/>
    <auditLog subsystem="SV" hostName="server_2" action="DELETE" description="Test deletion server 2."
username="admin" serviceDescription="network_users"/>
</auditLogs>
```

Here is a JSON example post data creating an audit log, (the wrapping object with the `auditLogs` array member is always required, even for a single audit log object):

```
{
  "auditLogs": [ {
    "subsystem":"Console",
    "hostName":"server_1",
    "action":"MODIFY",
    "description":"Test modification server 1.",
    "username":"admin"
  } ]
}
```

17.2.5 HTTP Status Codes

Code	Description
200	Zero or more audit logs were created synchronously or an asynchronous create started
401	Authentication/authorization error occurred
500	An internal server error occurred while creating audit logs or starting an asynchronous create

17.2.6 Example Create Responses

Audit log create responses do not include result entries with audit log record URIs. Audit logs are read-only and are not accessed individually.

Here are XML asynchronous and synchronous create responses:

```
<results successful="1" failed="0" entityType="AuditLog Async" operation="Insert" warning="0" count=
"1">
    <result><entity>1411761803932</entity><message>Job 1411761803932
submitted</message><status>success</status></result>
</results>
```

```
<results successful="2" failed="0" entityType="AuditLog" operation="Insert" warning="0" count="2">

<result><entity>com.groundwork.collage.model.impl.AuditLog@1ce1a1ba[auditLogId=97,subsystem=SV,hostName=s
addition server 2.,username=admin,timestamp=Fri Oct 24 11:40:45 MDT
2014,serviceDescription=local_processes]</entity><message>AuditLog
saved</message><status>success</status></result>

<result><entity>com.groundwork.collage.model.impl.AuditLog@2140c926[auditLogId=98,subsystem=SV,hostName=s
deletion server 2.,username=admin,timestamp=Fri Oct 24 11:40:45 MDT
2014,serviceDescription=network_users]</entity><message>AuditLog
saved</message><status>success</status></result>
</results>
```

```
<results successful="1" failed="2" entityType="AuditLog" operation="Insert" warning="0" count="3">
    <result><entity>org.groundwork.rs.dto.DtoAuditLog@4429dfb4[auditLogId=null
,subsystem=SV,hostName=server_2,action=INCREMENT,description=Test increment server
2.,username=admin,timestamp=null,serviceDescription=local_processes]</entity><message>Failed to save
AuditLog: No enum constant
com.groundwork.collage.model.AuditLog.Action.INCREMENT</message><status>failure</status></result>

<result><entity>com.groundwork.collage.model.impl.AuditLog@57238410[auditLogId=99,subsystem=SV,hostName=s
deletion server 2.,username=admin,timestamp=Fri Oct 24 11:59:45 MDT
2014,serviceDescription=network_users]</entity><message>AuditLog
saved</message><status>success</status></result>
    <result><entity>org.groundwork.rs.dto.DtoAuditLog@5e4ae40a[auditLogId=null
,subsystem=SV,hostName=server_2,action=ADD,description=Test add server 2.,username=null,timestamp=null
,serviceDescription=network_users]</entity><message>Failed to save AuditLog:
com.groundwork.collage.exception.CollageException:
org.springframework.dao.DataIntegrityViolationException: not-null property references a null or
transient value: com.groundwork.collage.model.impl.AuditLog.username; nested exception is
org.hibernate.PropertyValueException: not-null property references a null or transient value:
com.groundwork.collage.model.impl.AuditLog.username</message><status>failure</status></result>
</results>
```

Here are JSON asynchronous and synchronous create responses:

```
{
  "successful" : 1,
  "failed" : 0,
  "entityType" : "AuditLog Async",
  "operation" : "Insert",
  "warning" : 0,
  "results" : [ {
    "entity" : "1411761805028",
    "status" : "success",
    "message" : "Job 1411761805028 submitted"
  } ],
  "count" : 1
}
```

```
{
  "successful" : 2,
  "failed" : 0,
  "entityType" : "AuditLog",
  "operation" : "Insert",
  "warning" : 0,
  "results" : [ {
    "entity" :
"com.groundwork.collage.model.impl.AuditLog@7aec81d5[auditLogId=94,subsystem=SV,hostName=server_2,action=
addition server 2.,username=admin,timestamp=Fri Oct 24 11:40:43 MDT
2014,serviceDescription=local_processes]",
    "status" : "success",
    "message" : "AuditLog saved"
  }, {
    "entity" :
"com.groundwork.collage.model.impl.AuditLog@3d4005ff[auditLogId=95,subsystem=SV,hostName=server_2,action=
deletion server 2.,username=admin,timestamp=Fri Oct 24 11:40:43 MDT
2014,serviceDescription=network_users]",
    "status" : "success",
    "message" : "AuditLog saved"
  } ],
  "count" : 2
}
```

```
{
  "successful" : 1,
  "failed" : 2,
  "entityType" : "AuditLog",
  "operation" : "Insert",
  "warning" : 0,
  "results" : [ {
    "entity" : "org.groundwork.rs.dto.DtoAuditLog@2776182b[auditLogId=null
,subsystem=SV,hostName=server_2,action=INCREMENT,description=Test increment server
2.,username=admin,timestamp=null,serviceDescription=local_processes]",
    "status" : "failure",
    "message" : "Failed to save AuditLog: No enum constant
com.groundwork.collage.model.AuditLog.Action.INCREMENT"
  }, {
    "entity" :
"com.groundwork.collage.model.impl.AuditLog@14805c1[auditLogId=91,subsystem=SV,hostName=server_2,action=D
deletion server 2.,username=admin,timestamp=Fri Oct 24 11:38:46 MDT
2014,serviceDescription=network_users]",
    "status" : "success",
    "message" : "AuditLog saved"
  }, {
    "entity" : "org.groundwork.rs.dto.DtoAuditLog@7c129e3e[auditLogId=null
,subsystem=SV,hostName=server_2,action=ADD,description=Test add server 2.,username=null,timestamp=null
,serviceDescription=network_users]",
    "status" : "failure",
    "message" : "Failed to save AuditLog: com.groundwork.collage.exception.CollageException:
org.springframework.dao.DataIntegrityViolationException: not-null property references a null or
transient value: com.groundwork.collage.model.impl.AuditLog.username; nested exception is
org.hibernate.PropertyValueException: not-null property references a null or transient value:
com.groundwork.collage.model.impl.AuditLog.username"
  } ],
  "count" : 3
}
```