

# Grouper deprovisioning getting started

<a href="#">Wiki Home</a>	<a href="#">Download Grouper</a>	<a href="#">Grouper Guides</a>	<a href="#">Community Contributions</a>	<a href="#">Developer Resources</a>	<a href="#">Deployment Guide</a>
---------------------------	----------------------------------	--------------------------------	---	-------------------------------------	----------------------------------

## Get the software

Get Grouper 2.4+ (when released), or Grouper 2.3.0 patch API 107 and UI 44.

## Identify affiliations

Decide on affiliations to be deprovisioned. Probably start with one, e.g. "employee", or "member". You could pick multiple. You can also add more later.

## Configure grouper.properties

```

1. #####
   ## Deprovisioning
   #####

   # if deprovisioning should be enabled
   deprovisioning.enable = true

   # comma separated affiliations for deprovisioning e.g. employee, student, etc
   # these need to be alphanumeric suitable for properties keys for further config or for group extensions
   deprovisioning.affiliations =

   # Group name of the group that identifies generally if an entity is
   # in this affiliation. So if a group is deprovisioned
   # by various affiliations, then only deprovision if the entity in the group
   # is not in any affiliation eligible group.
   # e.g. VPN is deprovisioned by affiliations employee and student. If the person
   # is no longer an employee, but is still
   # a student, then dont deprovision.
   # deprovisioning.affiliation_<affiliationName>.groupNameMeansInAffiliation = a:b:c
   # deprovisioning.affiliation_employee.groupNameMeansInAffiliation = community:employee

   # folder where system objects are for deprovisioning
   # e.g. managersWhoCanDeprovision_<affiliationName>
   # e.g. usersWhoHaveBeenDeprovisioned_<affiliationName>
   deprovisioning.systemFolder = $$grouper.rootStemForBuiltinObjects$$:deprovisioning

   # autcreate the deprovisioning groups
   deprovisioning.autocreate.groups = true

   # users in this group who are admins of a affiliation but who are not Grouper SysAdmins, will be
   # able to deprovision from all grouper groups/objects, not just groups they have access to UPDATE/ADMIN
   deprovisioning.admin.group = $$deprovisioning.systemFolder$$:deprovisioningAdmins

   # number of days in deprovisioning group. Should be the amount of time for systems of record to catch
   # up and
   # for people to change external systems of record in manual processes
   deprovisioning.defaultNumberOfDaysInDeprovisioningGroup = 14

   #number of groups shown in the body of deprovisioning email
   deprovisioning.email.group.count = 100

   #deprovisioning reminder email subject
   deprovisioning.reminder.email.subject = You have $groupCount$ groups that have suggested users to be
   deprovisioned

   #deprovisioning reminder email body (links and groups are added dynamically)
   deprovisioning.reminder.email.body = You need to review the memberships of the following groups. Review
   the memberships of each group and click: More actions -> Deprovisioning -> Members of this group have
   been reviewed
   deprovisioning.reminder.email.body.greaterThan100 = There are $remaining$ more groups to be reviewed.

```

## Deprovisioning managers

Identify the deprovisioning managers and add them to the managers group. e.g. if your grouper.rootStemForBuiltinObjects is "etc", and your deprovisioning affiliation is "employee", then the group would be:

```
etc:deprovisioning:managersWhoCanDeprovision_employee
```

If you don't identify managers, then your Grouper admins can deprovision people