

Dateizugriffsrechte

UNIX-Berechtigungen

Symbolische Notation

Die Standard UNIX-Dateisystemberechtigungen, die auch auf den im HPC verwendeten Linux-Betriebssystemen zum Einsatz kommen, unterscheiden zwischen drei Berechtigungen:

- read, abgekürzt mit „r“, für die Berechtigung, Dateien zu lesen
- write, abgekürzt mit „w“, für die Berechtigung, Dateien zu bearbeiten sowie neue Dateien anzulegen
- execute, abgekürzt mit „x“, für die Berechtigung, Dateien als Programm auszuführen sowie Verzeichnisinhalte anzusehen

Diese Berechtigungen können für drei verschiedene Benutzer(Gruppen) vergeben werden:

- user, abgekürzt mit „u“, für den Dateieigentümer
- group, abgekürzt mit „g“, für die Gruppe, der eine Datei gehört
- others, abgekürzt mit „o“, für alle anderen NutzerInnen

Somit besteht ein Berechtigungssatz aus 3x3 Zeichen. `rxwxr-x---` bedeutet zum Beispiel, dass der Eigentümer alle Rechte hat (Lesen, Schreiben und Ausführen), die Gruppe nur Lesen und Ausführen, und alle anderen NutzerInnen gar keine Berechtigungen haben.

Berechtigungen anzeigen

Mit dem Befehl `ls -l <Pfad>` lassen sich die Berechtigungen für eine Datei oder alle Dateien in einem Ordner anzeigen:

Berechtigungen mit `ls -l` anzeigen

Berechtigungen ändern

Mit den Befehlen `chmod` („change mode“, d.h. Berechtigungen ändern) und `chown` („change owner“, d.h. Dateieigentümer und Gruppe ändern) lassen sich die Berechtigungen einer Datei bearbeiten. Die Syntax lautet dabei wie folgt:

Syntax von `chown` und `chmod`

```
chown [-R] user:group Pfad [Pfad...]  
chmod [-R] u=rwx,g=rwx,o=rwx Pfad [Pfad...]
```

Der Parameter `-R` („recursive“) gibt dabei in beiden Fällen an, dass die Berechtigungen nicht nur auf die angegebenen Pfade, sondern auch auf Unterordner und Dateien darin angewendet werden sollen. Im Falle von `chown` ist für `user` und `group` der Benutzername auf dem HPC (i.d.R. Die Unikennung) und eine Gruppe (i.d.R. Ein Projektkürzel) anzugeben, den die Datei zugeordnet werden soll. Für `chmod` kann mit `u=...,g=...,o=...`, angegeben werden, welche Berechtigungen jeweils Eigentümer, Gruppe und Andere auf der Datei haben sollen. Es können bei beiden Programmen eine oder mehrere Dateien auf einmal angegeben werden.



Vorsicht mit Berechtigungen für Alle

Die Berechtigungen `o=...` sollten sparsam verwendet werden, da diese Berechtigungen für alle HPC-Nutzenden gelten. Insbesondere bei Vergabe von Schreibrechten bedeutet dies, dass alle HPC-NutzerInnen die Datei sogar bearbeiten und löschen können. Falls die Benutzer- und Gruppenberechtigungen nicht ausreichen, weil die Datei zum Beispiel mit einer anderen Gruppe geteilt werden soll, sind dafür ACLs zu verwenden (siehe unten).

Berechtigungsprobleme selbst lösen

In vielen Fällen genügt es, im Hauptverzeichnis einen rekursiven Befehl zu starten, der die Berechtigungen in allen Unterordnern und Dateien korrigiert. Dazu sollte man sich erst im Klaren sein, welchem Benutzer und welcher Gruppe die Dateien zugeordnet werden sollen. Im Falle eines gemeinsam genutzten Projektverzeichnis ist vermutlich die Benutzergruppe der Arbeitsgruppe sinnvoll. Mit dem Kommando ``groups`` lässt sich herausfinden, welchen Benutzergruppen der eigene Account angehört.

Anschließend sollten folgende Befehle genügen:

Berechtigungsprobleme korrigieren

```
group=<Name der Gruppe>
find /Pfad/zum/Verzeichnis -type f ! -perm -g=rwc
chown -R user:group /Pfad/zum/Verzeichnis
chmod -R 770 /Pfad/zum/Verzeichnis
```

In manchen Fällen sitzen die Berechtigungsprobleme so tief, dass nicht alle Dateien demselben Nutzer gehören. In diesem Fall treten `Permission denied`-Fehler auf. Das bedeutet, dass auch andere Nutzer der Arbeitsgruppe diese Kommandos ausführen müssen, bis alle Dateien wieder die richtigen Berechtigungen haben, da nur der Eigentümer einer Datei die Berechtigungen anpassen kann.

ACLs

Manchmal verhindern ACLs den korrekten Zugriff auf das Verzeichnis. Wenn sichergestellt ist, dass die Zuordnung über eine einzelne Gruppe und einen einzelnen Benutzer ausreicht, und erweiterte Berechtigungen nicht benötigt werden, können mit folgendem Befehl alle ACLs auf allen Dateien und Unterordnern eines Verzeichnisses gelöscht werden:

ACLs löschen

```
find /Pfad/zum/Verzeichnis | xargs mmdelacl
```

Auch hier kann es sein, dass mehrere NutzerInnen der Arbeitsgruppe dieses Kommando ausführen kann, da auch hier nur der Eigentümer einer Datei / eines Ordners die ACLs löschen kann.

ACLs (Access Control Lists)

Sowohl das GPFS als auch NFS unterstützen ACLs (Access Control Lists). Diese können dazu verwendet werden, um neben den Standard-Dateiberechtigungen (Besitzer, Gruppe, Andere) einzelne Dateien oder ganze Ordner nur für bestimmte Nutzer oder Nutzergruppen freizugeben (siehe dazu Abschnitt "**Dateien / Ordner mit einzelnen Nutzern freigeben**").

Um die ACLs einer Datei oder eines Ordners zu bearbeiten, stehen verschiedene Kommandozeilentools zur Verfügung. Auf dem Login-Knoten lassen sich die ACLs mittels folgendem Kommando editieren:

NFSv4 ACL setzen (Login Node)

```
nfs4_setfacl --edit $PATH
```

und abrufen:

NFSv4 ACL abrufen (Login Node)

```
nfs4_getfacl $PATH
```

Auf den Compute-Nodes und den Storage-Knoten sind die Tools

NFSv4 ACLs (Compute Nodes)

```
mmeditac1
mmputac1
mmgetac1
```

zu verwenden, die sich allerdings in ihrer Syntax von `nfs4_setfacl` unterscheiden, und eventuell nicht alle Funktionen abbilden. (Siehe dazu `man mmeditac1`, `man mmputac1`, `man mmgetac1`, oder https://www.ibm.com/support/knowledgecenter/en/STXKQY_4.2.1/com.ibm.spectrum.scale.v4r21.doc/b11adm_mmputac1.htm)



Verwendung des Storage-Knotens für ACLs

Der Login-Knoten greift nur über einen NFS-Export auf das GPFS zu, anstelle wie die Rechenknoten und Storage-Knoten nativen Zugriff zu haben. Daher ist die Steuerung der ACLs vom Login-Knoten aus nur eingeschränkt möglich. Es wird daher empfohlen, die Verwaltung der ACLs direkt über den [Storage-Knoten](#) vorzunehmen.

Syntax

Die Verwendung der ACLs unterscheidet sich auf dem Login-Knoten und den Compute-Knoten bzw. Storage-Knoten. Um für beide Varianten eine optimale Dokumentation zu gewährleisten, existiert für beide Systeme eine eigene Anleitung:

Titel	Ersteller	Geändert	
ACL-Syntax (Login-Knoten)	Rose, Lukas	Okt. 06, 2022	1
ACL-Syntax (Compute-Knoten, Storage-Knoten)	Rose, Lukas	Sept. 21, 2022	