FutureGrid Privileged Access

Policy

FutureGrid allows developers associated with the project to request privileged access at other FutureGrid sites. The requestor and the hosting site negotiate this access, with mediation from the Operations Committee if necessary. The only type of privileged access that we currently allow to non-virtualized systems is via *sudo*.

Such privileged access is requested and granted according to the procedure described next and using the template agreement at the end of this document.

Procedure

When privileged access (i.e., root or root equivalents) to physical/bare-metal FG resources (hereon referred as resources) at an FG site by FG researchers at the same or another FG site is necessary for FG to fulfill its mission the following procedures must be followed:

1. Access Request: The FG Principal Investigator of the requesting site must acknowledge and endorse the request and the requestor by communicating the request to the FG Principal Investigator of the hosting site. The request should be made by email. The request should describe and justify what is needed. The endorser should assess whether requestors have the expertise required to use privileged access in a competent and responsible manner.

2. Evaluation of Request: The hosting site evaluates the requests and either approves it, denies it, or asks for modifications. A denial or request for modifications must include the reasons for the denial or modifications. The FG Principal Investigator of the hosting site provides the response. The response should be made by email in a timely manner, but no less than one week. If access is not granted, the requestor may request the Operations Committee to review the matter between the requestor and hosting site.

3. Acknowledgement of Responsibilities: If a request is approved, the requestor and a representative of the hosting site must sign an agreement that describes the responsibilities of all parties. Both requestor and hosting site should retain a copy of this agreement. A template for this agreement appears later in this document and includes:

    a. General responsibilities of anyone receiving privileged access on FutureGrid.
    b. Site-specific responsibilities of someone receiving privileged access at the specific hosting site.
    c. Responsibilities of the requestor in this specific privilege access situation and any exceptions to the responsibilities listed in a) and b).
    d. Responsibilities of the hosting site to the requestor

    This template will be customized by the hosting site to reflect local policies and practices.

4. Enabling Privileged Access: If a request is approved, the FG site PI or his/her delegate will assign a leader and system staff who will create a Jira entry and provide an effective plan for providing the requested access while meeting security requirements. Requests for additional resources (hardware, software or human) that may be required to enable privileged access should be forwarded to the FG PI.

5. Review of Access: If a requestor is granted privileged access, the requestor and hosting site will periodically review whether that privileged access should continue. A suggested period for this review is 6 months, but should be selected based on the work that is the reason for the privileged access. It is the responsibility of the requesting site PI to communicate revocation of privileged access to the hosting site PI when a person leaves employment. Violation of privileged access agreements will result in immediate revocation of privileged access.

Privileged Access Agreement (Sudo)

FutureGrid Grantee Responsibilities

Violation of the following will result in immediate cancellation of privileged access and other penalties in accordance with the regulations of the hosting site and applicable state and federal laws.

The Grantee is not a system administrator at the hosting site and must not perform system administration tasks - the Grantee must contact a system administrator to request that such tasks be performed. For example, the Grantee must not:

- Modify log files
- Create accounts
- Change passwords on other accounts (including the root account)
- Grant privileged access to others
- Run network daemons
- Install or update software packages in standard locations (e.g. via yum)
- Modify operating system configurations

The Grantee must follow good sudo practices which we define as:

- Grantee is not allowed to use *sudo* to run a shell: e.g. "*sudo* bash", or to just become root via "su". Instead, the Grantee should execute each of their commands via 'sudo <command>'
- Grantee is not allowed to use *sudo* to "su" to another user.

The Grantee commits to using the following best practices:

- Grantee must cho*ose* an extraordinary password for his/her personal account. This is especial*ly* important because the password can be used to exercise certain root privileges.
- Grantee must take extraordinary care in protecting his/her password. Passwords should never be sent in the clear over the network (if this does occur, the password should be changed immediately). Passwords should never be emailed. Passwords should never be shared with others.
- If Grantee makes changes to the system, he/she must write down what was done.

Site-Specific Grantee Responsibilities

Hotel

All *sudo* privileges on Hotel will be for a pre-determined length of time and will only be extended when formally requested, justified, and approved.
Except in unusual circumstances, full root *sudo* will not be given. This means Grantee will only be allowed to run those specifically requested commands that have been approved – e.g., "sudo /etc/init.d/httpd restart". Should full root *sudo* be required and approved, the Hotel system administrators may remove services and functionality that could potentially affect all Hotel users – e.g.,

GPFS may be disabled and unmounted so other user's data isn't accidentally destroyed or altered. Such service and functionality disabling will be discussed and agreed upon by Grantee and the Hotel administrators before access is granted. Full root *sudo* will **not** be granted on any of the Hotel management or storage servers for this reason. Also, Grantee will be required to obtain a one time password (OTP) token from the Hotel administrators to use for the duration that full root elevated privileges are granted. This OTP token is to be returned when the elevated privileges expire. Even though Grantee has full root, all restrictions in the "**FutureGrid Grantee Responsibilities**" section still apply.

Case-Specific Grantee Responsibilities and Exceptions

List any additional responsibilities in addition to those above.

List any exceptions to the responsibilities specified above.

Granter Obligations and Responsibilities

The following support will be provided by the granter: *(add or delete items as needed)*

- Reinstallation of original OS
- Updating/patching of original OS
- Minor fixes to problems caused by grantee
- Recovery of compromised systems

Review of Access

The Grantor and Grantee will review whether privileged access should continue every 6 months. The first such review will be MM/YYYY. *(adjust this period or use other conditions as needed)*

Appendix

Access to FutureGrid systems is controlled by the following:

- *(india)* Indiana University, Greg Pike (gregpike@indiana.edu)
- *(bravo)* Indiana University, Greg Pike (gregpike@indiana.edu)
- *(hotel)* University of Chicago, Ti Leggett (leggett@ci.uchicago.edu)
- *(foxtrot)* University of Florida, Mauricio Tsugawa (tsugawa@acis.ufl.edu)
- *(sierra)* San Diego Supercomputer Center, Greg Pike (gregpike@indiana.edu)
- *(alamo)* Texas Advanced Computing Center, David Gignac (dgignac@tacc.utexas.edu)

Why we use *sudo* instead of giving out the root password:

Root is all powerful.
In Unix and Unix-like systems, system administration privileges are all or nothing. A user either has root access or not, and root access implies complete control of a machine. If the machine in question is used by more than one person, or root has access to other systems or user files, it is more acceptable to give some users partial root privileges.

The root user can hide all of their actions.
sudo logs every command run via *sudo*. Having a record of what's being done with *sudo* helps us diagnose problems with individual systems/processes and general configuration issues, as well as helping us identify needed improvements.

The root password gives you access to any command on a system.
Via its config file, *sudo* can give a user root access for particular set of commands. This also avoids the "all or nothing" effect, allowing us to give individual users more control over their machines and to help themselves out of common problems.

Prefixing each command with "*sudo*" allows system administrators to keep a handle on what people are doing.
It is much easier to diagnose a problem if we know what has been done to a machine to create (or attempt to address) a problem. Similarly, it is much easier for us to understand how you cleverly solved some problem if we have a record of the actions you performed.

System administrators use *sudo*, too.
Log trails are an important part of enabling multiple people to manage the complexities of a large system.

Parts of this document incorporate text from the Stanford University Privileged Access Policy located at URL http://www-cs.stanford.edu/computing-guide/security/privileged-access-policy