An SSH Key Management System

Managing the associations between users, their SSH keys and accounts on multiple systems

D. Arkhipkin, A. Shiryaev (Particle Physics Lab, JINR - Dubna) J. Lauret, W. Betts (Brookhaven National Lab)

Motivations:

-Intense cyber-security scrutiny at DOE facilities includes a heavy emphasis on knowing who is doing what.
-User accounts in our experimental area are not necessarily tied to a single user ("group accounts"). One critical example is root, but this also includes "operator" accounts.

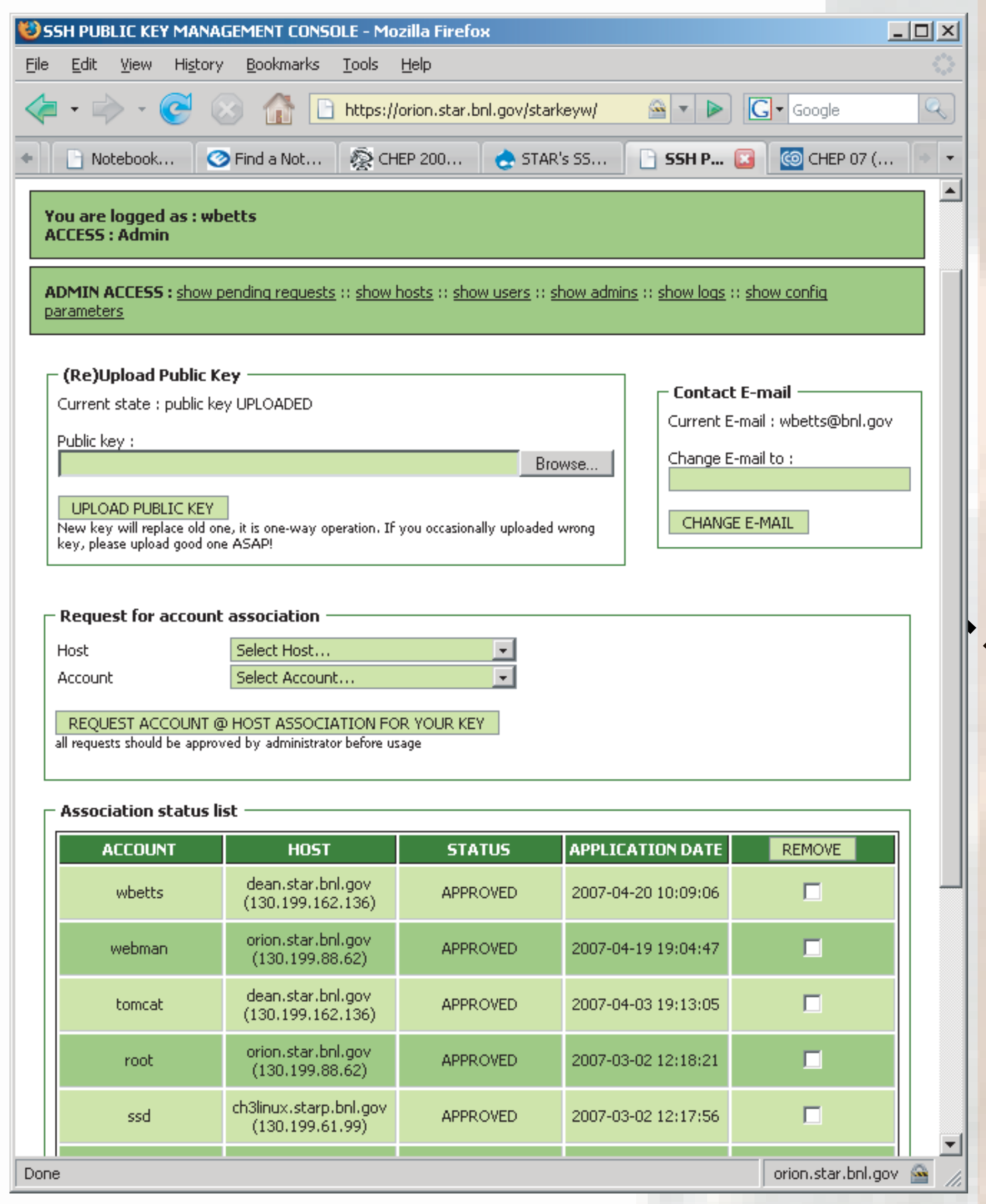
Requirements:

-Ability to manage ssh access to heterogeneous Linux hosts.
-Two factor authentication for remote logins (DOE requirement).
-Ability to identify individual user of each remote login to group accounts.
-Easy administration, including rapid ability to disable any users' access to all managed resources.

Major Features and Benefits:

- A wide variety of authentication protocols for web client access are easily implemented. (We use Kerberos.)
- Host certificates can be used to authenticate both the server and clients in the key distribution.
- Group account passwords can be changed for local logins without having to distribute them to remote users.
- A side benefit for users is the elimination of passwords to remember and type.

The users interact with the key management system entirely via web interface. The first time a user successfully logs in, he is automatically added to the database of registered users.


SSH PUBLIC KEY MANAGEMENT CONSOLE - Mozilla Firefox

https://orion.star.bnl.gov/starkeyw/

You are logged as : wbetts
ACCESS : Admin

ADMIN ACCESS : show pending requests :: show hosts :: show users :: show admins :: show logs :: show config parameters

(Re)Upload Public Key
Current state : public key UPLOADED
Public key :
UPLOAD PUBLIC KEY
New key will replace old one, it is one-way operation. If you occasionally uploaded wrong key, please upload good one ASAP!

Contact E-mail
Current E-mail : wbetts@bnl.gov
Change E-mail to :
CHANGE E-MAIL

Request for account association
Host: Select Host...
Account: Select Account...
REQUEST ACCOUNT @ HOST ASSOCIATION FOR YOUR KEY
all requests should be approved by administrator before usage

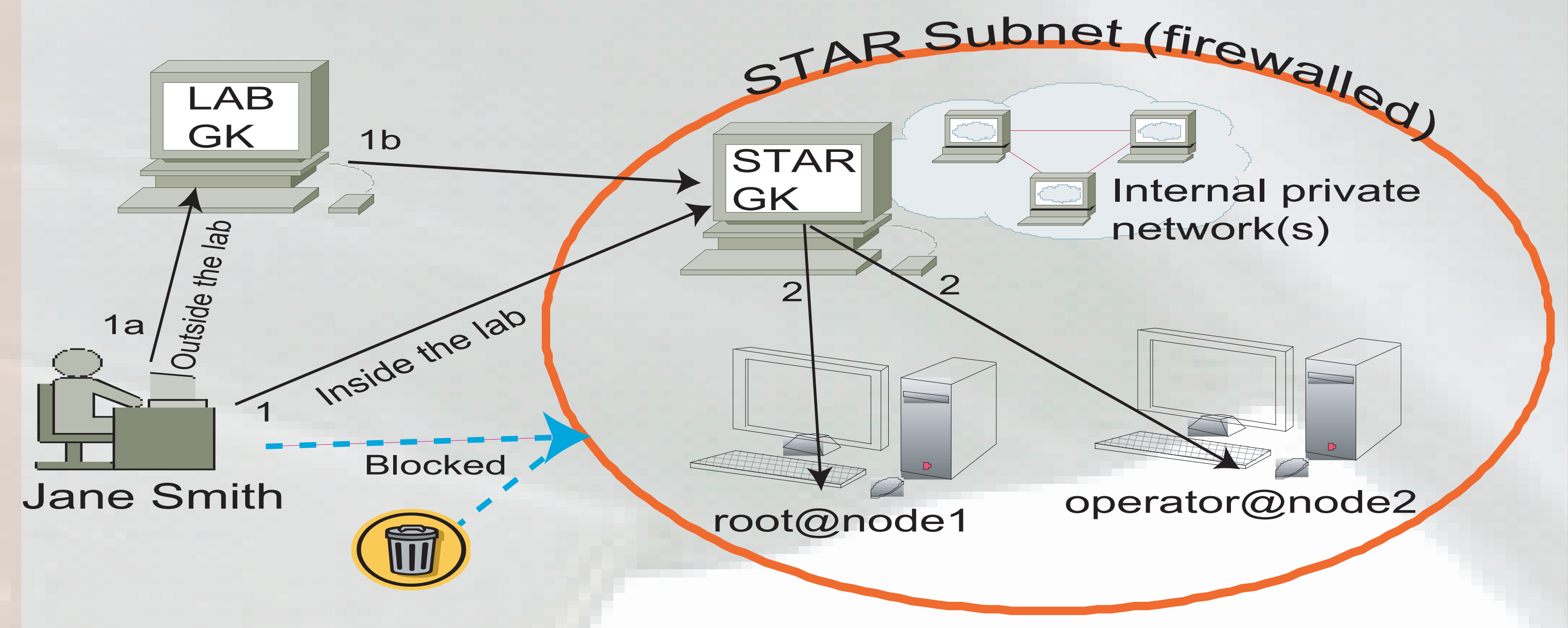
Association status list

ACCOUNT	HOST	STATUS	APPLICATION DATE	REMOVE
wbetts	dean.star.bnl.gov (130.199.162.136)	APPROVED	2007-04-20 10:09:06	☐
webman	orion.star.bnl.gov (130.199.88.62)	APPROVED	2007-04-19 19:04:47	☐
tomcat	dean.star.bnl.gov (130.199.162.136)	APPROVED	2007-04-03 19:13:05	☐
root	orion.star.bnl.gov (130.199.88.62)	APPROVED	2007-03-02 12:18:21	☐
ssd	ch3linux.starp.bnl.gov (130.199.61.99)	APPROVED	2007-03-02 12:17:56	☐

Done — orion.star.bnl.gov

Administrators use the same interface, but with management functions enabled.

Jane Smith logs into a STAR GK using her SSH keys from a remote location (1) or (1a+1b) depending on her location. (The STAR GK is also a client of the key management system). Key forwarding would then allow passwordless login to approved accounts on key management client nodes (2).



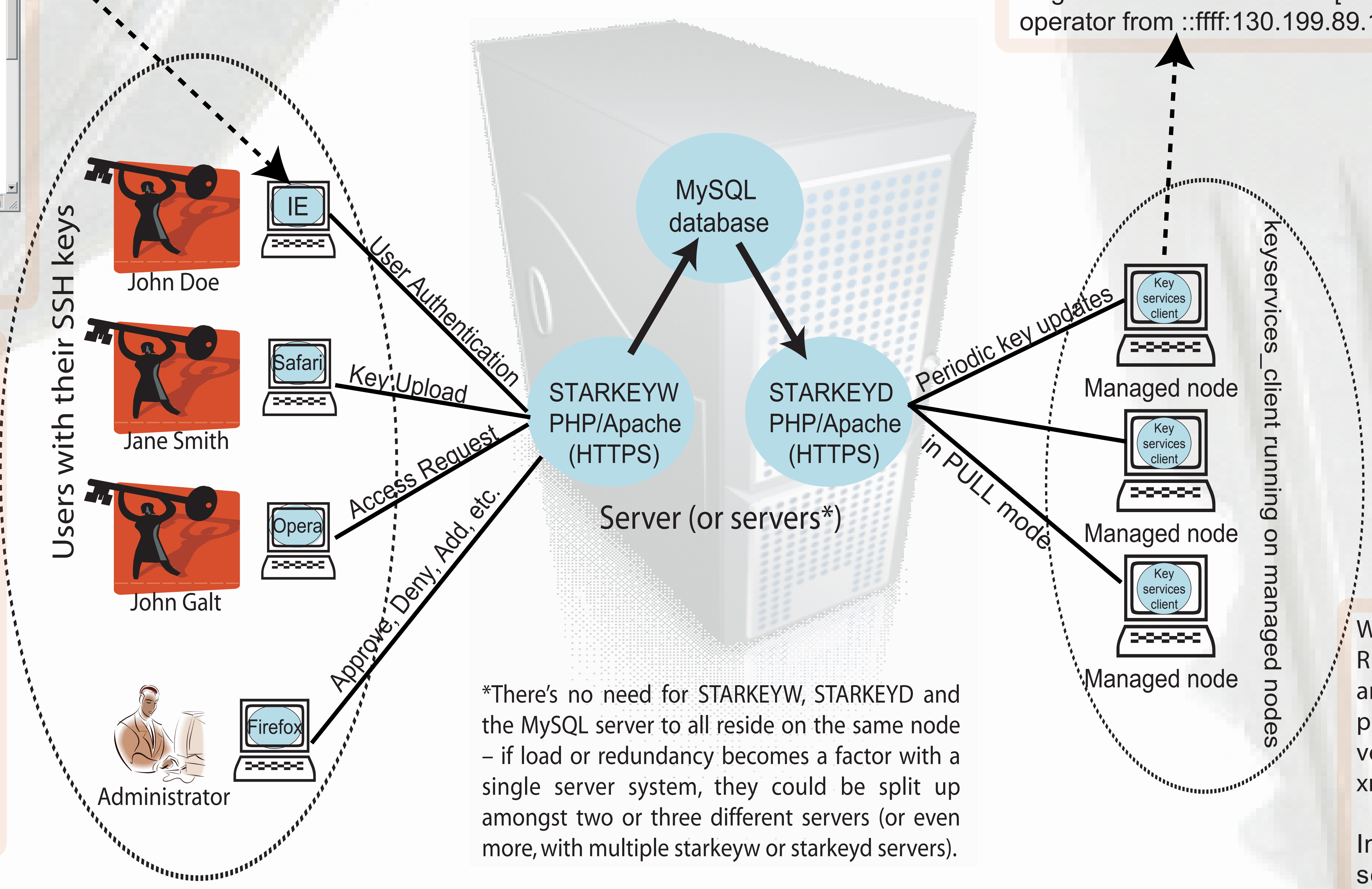
The STAR GK could also be dual homed on internal private networks to allow access to those nodes as well.

A client's SSH server is configured (`sshd_config`) with:
LogLevel VERBOSE
PasswordAuthentication no

User John Doe's key is added to the account he requested (eg. 'operator') by the keyservices_client `~operator/.ssh/authorized_keys`:
ssh-dss AAAAB3NzaC1kc3+...

`/var/log/secure` records his key fingerprint when he logs in as operator, so we know it is him and not Jane Smith:

```
Aug 27 16:40:58 orion sshd[4458]: Found matching RSA key: 9f:bb:46:e1:3f:95:d0:14:34:3b:0a:fd:98:c3:db:6b
Aug 27 16:40:58 orion sshd[4458]: Accepted publickey for operator from ::ffff:130.199.89.156 port 1318 ssh2
```



*There's no need for STARKEYW, STARKEYD and the MySQL server to all reside on the same node – if load or redundancy becomes a factor with a single server system, they could be split up amongst two or three different servers (or even more, with multiple starkeyw or starkeyd servers).

Typical usage sequence:

1. A sysadmin of a machine named FOO creates a user account named "operator" and, if not done already, installs the `keyservices_client` software on FOO.
2. A key service admin creates an "operator" account associated with host 'FOO' in the key Management system admin interface.
3. John Doe uploads (via the web) his or her public ssh key (in openssh format).
4. John Doe requests (via the web) that his key be added to operator's `authorized_keys` file on FOO.
5. A key service admin approves the request, and the `keyservices_client` places the key in `~JDOE/.ssh/authorized_keys`.
6. John Doe logs in as operator@FOO with his key.

Our documentation for users is available at:
http://drupal.star.bnl.gov/STAR/comp/onl/tools/sshkeymngt

We have `keyservices_client` RPMs for RHEL/SL 3 & 4 with 32-bit and 64-bit systems. The primary prerequisites for the client are PHP version >= 4.4.4 or 5.1.6 and xmlrpc-c >=1.06.08.

Installation prerequisites and setup instructions for all components are available at:
http://se51-63.jinr.ru/skd_setup/

CHEP '07, Sept. 2007 — Office of Science U.S. Department of Energy — Victoria, BC