Online Systems Tutorial

24-Jan-2003

S. Fuess
Contents

- Online subsystems
  - Complicated figure, but useful for reference
- Node, disk assignments
- Accounts
- Access controls
- Monitoring
  - Big Brother
- Web servers
- Control Room consoles
Assignments

- Node assignments
  http://www-d0online.fnal.gov/www/sys/operations/node_assignments.txt

- Disk assignments
  http://www-d0online.fnal.gov/www/sys/operations/disk_assignments.txt
Accounts

- Two important factors:
  - Authorization – that an account is present for a user on a node
    - Granted to any DO user with need
    - Access to group account may be sufficient
  - Authentication – that one can demonstrate knowledge of a password
    - The only allowed mode of access originating from outside of the Online system is by
      - Kerberos, for UNIX systems
      - NTLMv2 for Windows systems
        - eventually – Windows users should be aware of pending changes
Accounts

- On the “interactive” (Control Room, Monitoring, Host) systems
  - Authorization
    - Local accounts for system use only
    - NIS accounts for personal and group users
      - NIS domain server is d0olcluster
      - Personal accounts are “locked out” from non-Kerberos authentication
  - Authentication
    - Only root account has a local password
      - Kerberos .k5login access for remote logins
        - Personal Kerberos credentials (i.e. user@FNAL.GOV)
    - Group NIS accounts
      - NIS password only for local logins
      - Kerberos .k5login access for remote logins
        - Personal Kerberos credentials (i.e. user@FNAL.GOV)
        - Keytab Kerberos credentials (i.e. d0cap/d0/d0ol04.fnal.gov@FNAL.GOV)
    - Personal NIS accounts
      - Kerberos password for local logins (on most nodes)
      - Kerberos or .k5login access for remote logins
        - If a .k5login exists, then must include own credentials
Accounts

• On the “DAQ” (Readout, Level 2, Level 3) systems
  ♦ Authorization
    ▲ Local accounts for system, DAQ, and expert users
  ♦ Authentication
    ▲ Only root account has (should have) a local password
      – Kerberos .k5login access for remote logins
        • Personal Kerberos credentials (i.e. user@FNAL.GOV)
    ▲ DAQ local accounts
      – Kerberos .k5login access for remote logins
        • Personal Kerberos credentials (i.e. user@FNAL.GOV)
        • Keytab Kerberos credentials (i.e. d0run/d0/d0ol07.fnal.gov@FNAL.GOV)
    ▲ Expert user local accounts
      – Kerberos or .k5login access for remote logins
        • Personal Kerberos credentials (i.e. user@FNAL.GOV)
Accounts

- On the Controls systems
  - Authorization
    - Local accounts for expert users
  - Authentication
    - Expert user local accounts
      - Local password for local login

- No Kerberos! Remote logins are not allowed, and blocked by Online router
Some useful commands

- To check group account access, e.g.
  ```
  cat /home/d0cap/.k5login
  ```

- To see if a user has an NIS account, e.g.
  ```
  ypcat passwd | grep fuess
  ```

- To remotely log in to group account on an Online node, e.g.
  ```
  kinit fuess
  ssh -l d0cap d0ol04
  ```

- To log in to another node from a group account, e.g. as d0run
  ```
  setup d0online
  d0ssh -l d0cap d0ol04
  ```
Access controls

- Essential components of the computer security plan for the Online system are that:
  
  - The detector can operate with the Online system completely isolated from the external world
    - Well-defined isolation points
    - Can isolate from Offline, FCC, or both
    - Local versions of essential services
      - DNS server
      - KDC
    - Sufficient space to buffer event data for > 24 hours
  
  - Network access to the Online system is tightly controlled
    - Enforced by router module in Online switch acting as a “firewall”
    - Policy is “default deny”

  All this leads to functional limitations and operational confusion…
Access controls

Access lists are set for each VLAN (subnet) boundary *for each direction of network packets*

Format example:

[permit|deny] protocol source-host [eq port] destination-host [eq port]

“Incoming” examples:

permit tcp any any eq 22
permit tcp host odsoem host d0ola eq 1521

“Outgoing” examples:

permit ip any any reflect allow-231-out
• “Reflective” access
  ♦ Allowed outgoing packets create a temporary hole in the firewall, allowing return traffic between the specific node/port pairs
    ▲ Lifetime of 5 minutes
    ▲ Lifetime reset on each outgoing packet

  ♦ Normal example: Online access to Offline web page
    ▲ Web client on dynamically assigned port \( \rightarrow \) port 80 on web server
      – Opens hole for return traffic
    ▲ Port 80 from web server \( \rightarrow \) web client port
    ▲ Each new request “reopens” the hole
      – Note: automatically updating page will work, as tcp acknowledgement packet will reset timer
“Reflective” access (cont’d)

- Catch #1: telnet from Online to Offline
  - telnet client on dynamically assigned port → port 23 on Offline server
    - Opens hole for return traffic
  - Port 23 from Offline server → telnet client port
    - Allowed *within timeout period* for return traffic
  - If > 5 minutes inactivity, then initiate activity from Online client side
    - Works, opens new hole
  - If > 5 minutes inactivity, but then new activity from Offline server side
    - Blocked! Hole has expired
    - For example, output from long-running program on Offline server

Solution: there is none
• “Reflective” access (cont’d)
  ✭ Catch #2: telnet from Offline to Online
    ▲ Offline telnet client on dynamically assigned port → port 23 on Online server
      – Explicitly allowed in Access Control Lists (ACLs)
    ▲ Port 23 from Online server → Offline telnet client port
      – Explicitly allowed in ACLs
    ▲ Start X application on Online server:
      Online X client on dynamically assigned port → port 6000 on Offline X server
      – Works, opens new hole for return traffic
    ▲ Port 6000 from Offline X server → Online X client port
      – Allowed *within timeout period* for return traffic
    ▲ If > 5 minutes inactivity, then initiate activity from Offline (X server) side
      – Blocked! Hole has expired
      – For example, attempting input into GUI

Solution: tunnel X through an ssh connection
Access controls

- “Reflective” access (cont’d)
  - Recommendation: ssh from Offline to Online (configured to forward X!)
    - Offline ssh client on dynamically assigned port → port 22 on Online server
      - Explicitly allowed in Access Control Lists (ACLs)
    - Port 22 from Online server → Offline ssh client port
      - Explicitly allowed in ACLs
    - Start X application on Online server:
      Online X client on dynamically assigned port → port 6010 on Online
      Accepted by Online sshd daemon, forwarded to Offline ssh client port
      - Through explicitly allowed ssh hole already in use
    - Received by Offline ssh client, forwarded to Offline port 6000 (X server)
    - All subsequent X communication tunneled through open ssh hole
Online Tutorial 24-Jan-03

Access controls

- Some useful commands
  - To see if X forwarding is on by default (UNIX)
    cat /etc/ssh_config
      ▲ Should see
      ForwardX11 yes
  
  - To see if X forwarding is on for own account (UNIX)
    cat ~/.ssh/config
      ▲ Should see
      ForwardX11 yes
  
  - To check that X is forwarded:
    Echo $DISPLAY
      ▲ Should see the *remote* node with a server number 10 or higher
Monitoring – Big Brother

Big Brother main display

*click buttons for more info*

http://www-d0online/bb
Monitoring – Big Brother

Summary display

(click button)
Monitoring – Big Brother

Big Brother larrd display
CPU, memory, disk usage
Monitoring – Big Brother

Big Brother topp display

Warning: all BB updates are synchronized, so often report themselves as current major user!
Monitoring – Big Brother

Big Brother disk display

Local disk usage

See d0ola/b/c for cluster disks
Monitoring – Big Brother

Big Brother `procs` display

```plaintext
httpd >1 - not running
inetd >=1 - 2 instances running
missed >=1 - 1 instance running
afsd >=1 - 1 instance running
mounted >=1 - 1 instance running
ypserv >=1 - 1 instance running
update >=1 - 1 instance running
kioadsv >=1 - 1 instance running
init >=1 - 2 instances running
ibus >=1 - 12 instances running
xmt >=1 - 1 instance running
send >=1 - 0 instances running
sendmail >=1 - 1 instance running
smtpd >=1 - 1 instance running
smtp >=1 - 1 instance running
portmap >=1 - 1 instance running
blogin >=1 - 1 instance running
syslog >=1 - 1 instance running
```
Web Servers

- There are several “internal” and “external” servers
  - Internal: visible only from within Online system
  - External: visible from anywhere
- One strategy is to mount / display from *same* disks
  - NFS mounted from a central server
  - Read-only mount to external servers
  - Appropriate ACL holes in router
  - Internal server:
    - http://www-d0ol.fnal.gov (alias for d0ol01)
  - External server:
    - http://www-d0online.fnal.gov (alias for d0online2)
- Other strategy is for server to act as client of internal node
  - Appropriate ACL holes in router
  - External server:
    - http://www-d0l3mon.fnal.gov
Control Room consoles

- Linux provides, by default, 6 serial and 1 graphical sessions
  - Graphical session is default
  - Switch among them with CTRL-ALT-F1 through CTRL-ALT-F7 keys
    ▲ CTRL-ALT-F7 is the graphical session
- X is the windowing system for Linux
  - As opposed to Windows, where X has to be run “on top of” the native windowing system
  - The windowing system is the function of the “X server”
    ▲ /etc/X11/X
      - Configured by /etc/X11/XF86config-4
        ▪ Sets properties of graphics cards and monitors
      - Manages the DISPLAYs
      - Restart with CTRL-ALT-BACKSPACE – logs you out!
  - The X “display manager” runs to manage graphical logins
    ▲ /usr/X11R6/bin/xdm
  - The X “window manager” runs upon login; we use fvwm
    ▲ /usr/X11R6/lib/X11/fvwm2
      - Configured to set virtual windows, menus, etc
      - Restartable “hot” from menu