

DØ Online Computing

Access Methods

Version 1.3	S. Fuess	23-Nov-2001	Add windows access info
Responsible person	S. Fuess	fuess@fnal.gov	630-840-2452

1. Introduction

1.1 Purpose

The purpose of this document is to describe the methods for console and network access to the DØ Online computing systems. The DØ Online is a Critical System, with a number of access restrictions imposed. In particular, external access to the Online system must be made using Kerberos authentication.

1.2 Guidance

This document will attempt to address the access needs of the users of the Online computers. In any situation there are likely to be a number of possible solutions. The focus will be on a perceived best solution. Wherever possible solutions employing Kerberos authentication and ssh encryption will be suggested.

1.3 Background

The DØ Online computers have both personal and group accounts. Only personal accounts are allowed to have Kerberos principals, hence external access via a group account is not possible. However, local console access to group accounts is possible. Personal accounts will not have local passwords.

1.4 Associated document

A companion document, *DØ Run II Online Computing Network: Access Controls*, describes in complete detail the VLAN structure of the Online network and the access controls imposed between VLAN elements and the external network.

2. Access to UNIX nodes

2.1 Console access to UNIX nodes

The console login program on the Fermi Red Hat Linux (FRHL) systems does not currently accept Kerberos authentication. Therefore it is only possible to log in locally with an account with a local password. This includes the group and system accounts.

Note: Use a group account with its well-known password to log on locally.

Example: On the d0o104 (the Captain's console) login window supply the d0cap account name and password.

2.2 Remote access to UNIX nodes from a UNIX node

2.2.1 From a private account

If Kerberos credentials have not already been obtained on the local machine, use `kinit` to create a credential.

- `kinit <user>`

Example: `kinit fuess`

Warning: Do not do a kinit over the network unless you are sure all sessions are encrypted!

Then select a remote access method (omit the `command` option to get a login session).

- `ssh [-l <user>] <host> [command]`

Example: `ssh d0o104 "ls -l /tmp"`

- `rsh [-l <user>] <host> [command]`

Example: `rsh d0o104 "ls -l /tmp"`

- `scp [<host1>:]<filename1> [<host2>:]<filename2>`

Example: `scp d0o104:junk1 d0o105:junk2`

2.2.2 From a private account to a group account

If Kerberos credentials have not already been obtained on the local machine, use `kinit` to create a credential.

- `kinit <user>`

Example: `kinit fuess`

The private account principal must be listed in the `.k5login` file of the group account. Then select a remote access method (omit the `command` option to get a login session).

- `ssh -l <group user> <host> [command]`

Example: `ssh -l d0run d0o104 "ls -l /tmp"`

- `rsh -l <group user> <host> [command]`

Example: `rsh -l d0run d0o104 "ls -l /tmp"`

- `scp [[<user>@]<host1>:]<filename1> [[<user>@]<host2>:]<filename2>`

Example: `scp d0o104:junk1 d0run@d0o105:junk`

Note: You do not need a private Online account to access an Online group account. You merely need to have your principal listed in the `.k5login` file of the group account.

2.2.3 From a group account to the same or new group account

If Kerberos credentials have not already been obtained on the local machine, first use `kinit` with the project principal name to create a credential. Use the `keytab` file appropriate for the account and the node (see Section 3). The project principal must be in the `.k5login` file of the group account on the remote node. Note that it is not possible to switch to a user account with this method.

- `kinit -k -t /var/adm/krb5/<user>_keytab <user>/d0/<host>.fnal.gov`

Example: `kinit -k -t /var/adm/krb5/d0run_keytab d0cap/d0/d0o104.fnal.gov`

- `ssh -l <group user> <host> [command]`

Note: “-l <group user>” is required with ssh from a group account

Example: `ssh -l d0cap d0o107 "ls -l /tmp"`

- `rsh [-l <group user>] <host> [command]`

Example: `rsh d0o107 "ls -l /tmp"`

- `scp [[<user>@]<host1>:]<filename1> [[<user>@]<host2>:]<filename2>`

Example: `scp d0run@d0o107:junk1 d0lum@d0o105:/home/d0lum/junk2`

2.2.4 From a group account to a private account

If one is logged into a group account because they are sitting at a Control Room console, to access the same or another UNIX node as a private user one must use kinit to get new credentials. Be aware of the many options in kinit to set ticket forwarding and lifetime.

- `kinit <user>`

Warning: Do not do a kinit over the network unless you are sure all sessions are encrypted!

Suggestion: do private work in the Personal area on the Control Room consoles. Remember to *kdestroy* before leaving.

With these personal credentials the usual access methods are available. The use of ssh is encouraged, as the tunneling of X connections will avoid router ACL interference.

- `ssh [-l <user>] <host> [command]`

Note: In this case the -l <user> is optional. Ssh picks up the user name from the principal name. For private accounts these are usually equivalent. For group accounts the project principals will also include /d0/<node>.fnal.gov.

2.3 Remote access to UNIX nodes from a Windows node

Windows nodes can be configured to access UNIX nodes by several methods. Remember that remote access is only possible using Kerberos authentication to a personal account on the Online system.

2.3.1 Using a Windows ssh client and a CryptoCard

The use of ssh is encouraged as the best means to tunnel X across router boundaries. There is no Windows ssh client which supports Kerberos, hence the only possible authentication mechanism is via a CryptoCard.

- Configure a ssh client (`ttssh` or `putty`). Check that X forwarding is enabled.
- Connect to the remote node. Supply a personal user account and a blank password.
- Answer the CryptoCard challenge.

Once logged in, change to a group account if desired. There are several methods, but ssh is preferred to maintain the ssh tunnel for X traffic.

- `ssh -l <group user> <host> [command]`

Example: `ssh -l d0cap d0o104`

The Kerberos credentials remain that of the original user until a subsequent kinit.

2.3.2 Using a Windows telnet client with a CryptoCard

A telnet client can also be used in a similar manner to an ssh client. The only difference is that any X application run will not tunnel its traffic through the ssh port. The router Access Controls will permit this X traffic, but with a 5-minute idle timeout. The use of ssh is recommended.

2.3.3 Using a Windows telnet client with WRQ or Kerberized Exceed7

This procedure is similar to the preceding, but requires either a prior local Kerberos authentication or an authentication step during the connection process. Again, any X application is subject to having its connection broken by the router Access Controls.

3. Remote scripts

3.1 Using group accounts

With the tightening of the Kerberos belt (noose?) on the Online nodes, many of the methods used in the past to execute remote scripts or start remote applications must be "enhanced". The basic issue is that any node-to-node activity will need to have Kerberos authentication. No longer will `.rhosts` or `.shosts` files allow you to do things.

Further complicating this is that most of these activities take place in the group accounts, and not in private user accounts. Since the group accounts have no default Kerberos principals, we have to use *service* or *project* principals.

These principals are maintained in *keytab* files, specific to each machine. The specific group account has a *keytab* entry that allows the creation of a service principal. Only the group account has permission to access the specific *keytab* file. Remember, to "be in the group account" means you have physically logged in at the Control Room or have come in via Kerberos authentication from a remote node.

To access the *keytab* file:

- `kinit -k -t <keytab file> <service principal>`

Example: `kinit -k -t /var/adm/krb5/d0run_keytab d0run/d0/d0o107.fnal.gov`

Note the elements of this:

- The *keytab* files are kept in `/var/adm/krb5` with the name convention of `<group user>_keytab`
- The principal is named `<group user>/d0/<node>.fnal.gov` where the Kerberos realm name suffix of `@FNAL.GOV` is assumed.

Contact me to request a keytab file. I have to ask the Computing Division for the creation.

To use this principal, it must be listed in the `.k5login` file of the remote node/user. By default we put this in the `.k5login` of the group account itself. Since this is in `/home`, mounted on all nodes, then you can use the principal to get to any node as the same user. To get access to a different group account, add the principal name to the other account's `.k5login`. Remember to use the long principal name in `.k5login`.

Example: `d0run/d0/d0o107.fnal.gov@FNAL.GOV`

Note how the node name of the *source* of the scripts is important. If you will run from several sources, you'll need a *keytab* file for each source and an entry in `.k5login` for each source.

Also note that when using ssh with the service principal one must remember to use the `"-l <user>"` option, otherwise ssh uses the principal name as the user name and would incorrectly append `/d0/<node>.fnal.gov`.

4. Access to Windows nodes

4.1 Console access to Windows nodes

Login at a Windows console requires a local or domain account and password. The d00lnt domain has the standard set of group accounts with the usual passwords (as well as can be manually synchronized).

4.2 Remote access to Windows nodes from within Online system

4.2.1 From a Linux system

4.2.1.1 Using vnc

The *vnc* product provides a method of viewing a remote Windows desktop on a local Linux console. See section 5.2 for *vnc* installation instructions. The following description assumes that *vnc* has been installed as a *ups* product, which is appropriate for the Online Linux systems.

On your local Linux node, perform the sequence of commands:

- `setup vnc`
- `vncviewer [-shared] [<host>[:<display#>]]`

You will be prompted for a remote node name and a password, which is associated with the *vnc* server installation on the remote node. If the connection is successful a window with the remote screen will become visible.

Hint: Use the F8 key for useful options such as “send ctrl-alt-del” or “close”.

The *vnc* viewer produces an X display that can be transported to a remote screen in the usual manner. If X is being forwarded via *ssh*, then the X traffic can pass the Online router boundary with no problems. However, X is not an efficient protocol for slow network links. In that case, a forwarding of the complete *vnc* protocol, described in Section 4.3.2, is more efficient.

4.2.1.2 Other methods

Some of the Online Linux systems have *vmware* installed. With *vmware* you can start a virtual Windows machine on the Linux node. Within the Windows virtual machine you can use any of the Windows-to-Windows communication tools, including *NetMeeting* and *Terminal Server*.

4.2.2 From a Windows system

4.2.2.1 Using vnc

The installation of *vnc* will produce Start menu items for the client and server. The *VNCviewer* client is started via the Start menu. You will be prompted for the server node and password.

Hint: Use the upper left window icon for useful options such as “send ctrl-alt-del” or “close”.

4.2.2.2 Other methods

From a Windows you can use any of the Windows-to-Windows communication tools, including *NetMeeting* and *Terminal Server*.

4.3 Remote access to Windows nodes from outside of Online system

4.3.1 Authentication and access control

The primary consideration in remote access of an Online node is proper authentication. As Kerberos is the only accepted authentication method, then the initial connection from external to Online nodes must be with a Kerberos-aware mechanism. In addition, since the Online router access controls permit only a few protocols (see the companion document, *DØ Run II Online Computing Network: Access Controls*), the remote access application must use permitted protocols or appropriately tunnel its protocols over an established connection.

4.3.2 Remote access using vnc and ssh tunneling

The principal advantage to using *vnc* for access to Windows nodes is that the protocol can be tunneled using *ssh* (*NetMeeting*, for example, can not be easily forwarded as both client and server use the same port number). There are several possible configurations; a simple one follows.

- Configure and start the *vnc* server on an Online Windows node. The server will listen on a port equal to 5900 plus the display number. Almost always the display number will be 0, so the port of interest is tcp:5900.
- Configure your local *ssh* client to forward local port 5900 to the remote node, also at port 5900. On a Windows client, *ttssh* allows such *ssh* port forwarding in the “Setup” / “SSH Forwarding” menu item. On a Unix client, this will be specified in the *ssh* command line or a configuration file.
- Use *ssh* to connect to an Online Unix system. This step is where the authentication to access the Online system occurs, plus establishes the *ssh* tunnel through which the *vnc* traffic will pass.
- Start the local *vnc* viewer application. Connect to *localhost* as the server location. The password will be that defined when starting the *vnc* server on the Online Windows node.
- See above for hints on operation of the *vnc* viewer.

4.3.3 From a Linux system

4.3.3.1 Using vnc

The *vnc* product provides a method of viewing an Online Windows desktop on an external Linux console. See section 5.2 for *vnc* installation instructions. The following description assumes that *vnc* has been installed as a *ups* product, which is appropriate for the Online Linux systems.

First, the *vnc* server must be established on the Online Windows node. See Section 5.2.2.2 for details.

On your external Linux node, perform the sequence of commands:

- `ssh [-l <user>] -L 5900:<Online Windows node>:5900 <host>`

Example: `ssh -L 5900:d0o1ntsvr2:5900 d0o104`

- `setup vnc`
- `vncviewer [-shared] [<host>[:<display#>]]`

You will be prompted for a remote node name (use *localhost*) and a password (which is associated with the *vnc* server installation on the Online Windows node). If the connection is successful a window with the remote screen will become visible.

Hint: Use the F8 key for useful options such as “send ctrl-alt-del” or “close”.

4.3.4 From a Windows system

The installation of *vnc* will produce Start menu items for the client and server. The *VNCviewer* client is started via the Start menu. You will be prompted for the server node and password.

Hint: Use the upper left window icon to get useful options such as “send ctrl-alt-del” or “close”.

4.3.5 Alternatives to vnc

Under development. We are looking at *ipsec* and/or *Windows Terminal Server*.

5. Appendix

5.1 *ttssh*

5.1.1 Introduction

The freeware program *ttssh* is a ssh client for Windows.

5.1.2 Installation

- Download *ttssh* from the url:
<http://www.zip.com.au/~roca/ttssh.html>
- Select the Teraterm Pro link and run from the current location
- This will run a WinZip self-extractor. Extract the files to a temporary folder, for example C:\temp\temp23.
- Locate and run setup.exe from the folder into which files were just extracted. The will install Teraterm Pro. Choose a location into which to install the programs, for example C:\Program Files\TTERMPRO.
- Go back to the web page and select the download and use link and the USA option. Run from the current location and extract the files with WinZip. **Extract the *ttssh* files to the same folder where you just installed TTERMPRO**, for example C:\Program Files\TTERMPRO.
- Locate the *ttssh.exe* file in the above folder. Create a shortcut (right click, drag, and select “Create Shortcut Here”) on the desktop or in a Start folder (right click on Start menu, select “Open”, and navigate to the appropriate folder).

5.1.3 Customization

The shortcuts created to link to the *ttssh* executable can easily be customized. Right click on the shortcut and select “Properties”.

- To set the shortcut to use ssh to a specific node, for example d00l19, modify the “target” value to:

"C:\Program Files\TTERMPRO\ttssh.exe" d00l19.fnal.gov:22 /ssh

- To set X forwarding (recommended!) append to the above:

/ssh-X

- To set forwarding for other ports, for example for vnc to d00lntsvr2, append to the above:

/ssh-L5900:d00lntsvr2.fnal.gov:5900

Note that many of the *ttssh* command line parameters can also be set from the Setup menu item and saved in the default or a specific configuration file.

5.2 vnc

5.2.1 Introduction

The *vnc* freeware package contains a client and a server allowing remote desktop control of Unix and Windows machines. We can use *ssh* to tunnel *vnc*, thus allowing remote connections to Windows nodes across the Online router boundary.

The *vnc* package is available on the web at:

<http://www.uk.research.att.com/vnc>

5.2.2 Installation

5.2.2.1 Linux systems

The *vnc* package has been placed in `/d0usr/products/vnc` on the Online system. Use the *ups* tools to put the product in the path.

- `setup vnc`

There are two principal binaries in the *vnc* package.

Server installation:

Xvnc is the server package. We will not normally run a *vnc* server on a Linux platform.

Client operation:

vncviewer is the client package. The `-shared` option requests a shared session. Start a viewer session with:

- `vncviewer [-shared] <host>[:<display#>]`

Note: `<host>` will be “localhost” if using *ssh* tunneling

5.2.2.2 Windows systems

The *vnc* package is available on the Online Windows nodes from:

<\\d0olnt\d0olntfs1\apps\vnc>

The installation of *vnc* will produce Start menu items for the client and server.

Server installation:

The *WinVNC* server is installed as a Windows service. It's operation can be controlled via the Control Panel / Services window. When the server is started a password will be requested; this is the password that clients must provide. Note that there is also an option for allowing shared sessions, so that multiple clients may simultaneously connect.

Client operation:

The *VNCviewer* client is started via the Start menu. You will be prompted for the server name and password. There is an option to request a shared session if desired.

Note: The remote server name will be “localhost” if using *ssh* tunneling