ctys-uc-**VNC**(7)
Use-Cases for **VNC**

November 22, 2010

Contents

.

1 General

The VNC plugin supports access to remote desktops by the NFB protocol. The access could be either by combination of provided client and server programs to a native target, or by utilizing the client only either to an application or hypervisor supoorting the NFB protocol.

The automated signon when connecting a vncviewer to a vncserver requires a password as supported by vncpasswd. In order to avoid any user interaction for password requests the password is stored into the passwd file in **$HOME/.vnc** and is set to a default "install". This has to be changed once installed.

The default session type is **VNC**, thus the '-t vnc' option could be omitted within the following examples. The call

```
ctys -t VNC -a create=l:test
```

is identical to

```
ctys -a create=l:test
```

This behaviour could be changed within the configuration file 'ctys-conf.sh' by the variable 'DEFAULT_C_SESSIONTYPE'. For future safety of scripts despite the pre-set default the session type should be provided explicitly.

2 Start a Local Desktop Session

This opens a local session, where the VNCserver as well as the VNCviewer are executed locally.

```
ctys -t VNC -a create=l:tst1
```

The "localhost" is hard-coded to behave as a sub-shell call too, thus the following call is internally handeled identical to the previous

```
ctys -t VNC -a createl=l:tst1  \$USER@localhost
```

This case is called **DISPLAYFORWARDING** which is almost the same as the X11 display forwarding.

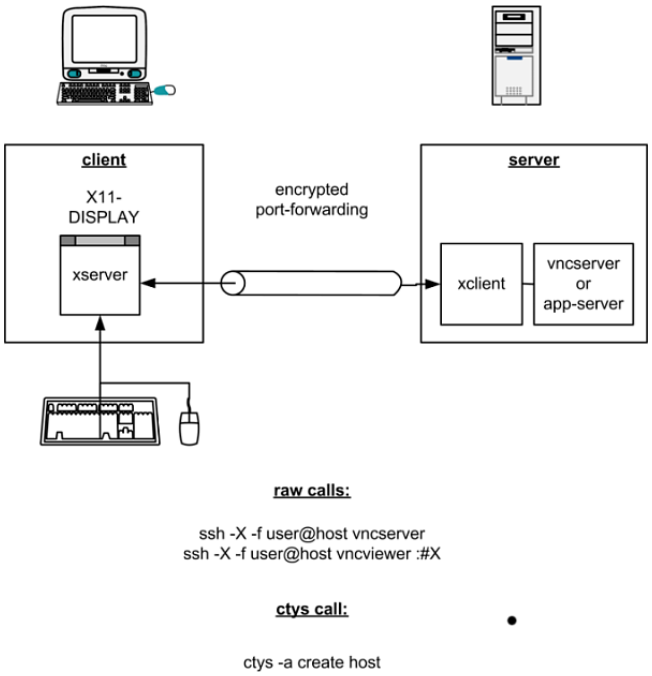

Figure 5: DISPLAYFORWARDING

3 Start a Remote Desktop Session

This call opens a remote desktop with DISPLAYFORWARDING, which is a coallocated VNCserver with a VNCviewer on the <execution-target>.

```
ctys -t vnc -a create=l:tst1 -L DISPLAYFORWARDINGF  <host>
```

The same could be written as:

```
ctys -t vnc -a create=l:tst1 -L DF lab00
```

The Client-Location "-L DISPLAYFORWARDING" is default for the original distribution, thus could be written as:

```
ctys -t vnc -a create=l:tst1 lab00
```

4 Start Bulk Desktop Sessions

This call opens 3 desktops on the remote host. The internal limit is set by default to 20.

```
ctys -t vnc -a create=bulk:3,l:tst lab00
```

The following call cancels all session by addressing their labels. The complete label is required here, which is an extended label by the incremental bulk-counter.

```
ctys -t vnc -a cancel=l:tst000,l:tst001,l:tst002 app2
```

The same function with usage of IDs.

```
ctys -t vnc -a cancel=i:2,i:3,i:4 app2
```

Current version supports as an implicit bulk addressing the keyword "ALL" only, which kills literally all VNC session where the appropriate permissions are available.

```
ctys -t vnc -a cancel=all app2
```

It should be recognized, that the CANCEL action is just a call to "vncserver -kill <display>" command, when this does not succeed, a "kill" will be placed. The clients are killed by UNIX-calls when required.
So the user is responsible for shutting down applications running within the CANCEL-ed sessions.

5 Start a Remote Desktop with a Local Client

In case of a "Remote Desktop with Local Client" the server is started on the given <execution-target>, whereas the client is locally started on the caller's machine. This structure is called CONNECTIONFORWARDING and requires beneath the client and server processes a third, the connecting encrypted tunnel. The tunnel is established by means of OpenSSH and used as the local peer for the Client. This whole procedure of starting the processes and the establishment of the tunnel is controlled and preformed by ctys.
The scenario differs in all steps except the start of the server process from the previously described DISPLAY-FORWARDING structure. In case of CONNECTIONFORWARDING the whole process is set up in three steps.

1. start of server process
2. establishment of the encrypted tunnel
3. start and connect the client process to the tunnel

The tunnel is established in the so called **one-shot mode**, where the connection is opened for an inital time period and closes automatically when the life-time threshhold is reached, or afterwards, when the server disconnects. The period of the initial lifetime is defined by the variable "SSH_ONESHOT_TIMEOUT", which is by default set to 20seconds.
The following call starts a remote server with a local client.

```
ctys -t vnc -a create=l:tst -L CF lab00
```

REMARK: When the error message **Authentication Failure** is replied and no client window occurs, the reason is the differing passwd files of VNC. For the remote client by the option '-L CF' - ConnectionForwarding - the local passwd file of VNC has to contain the same password as the remote machine running the vncserver process.

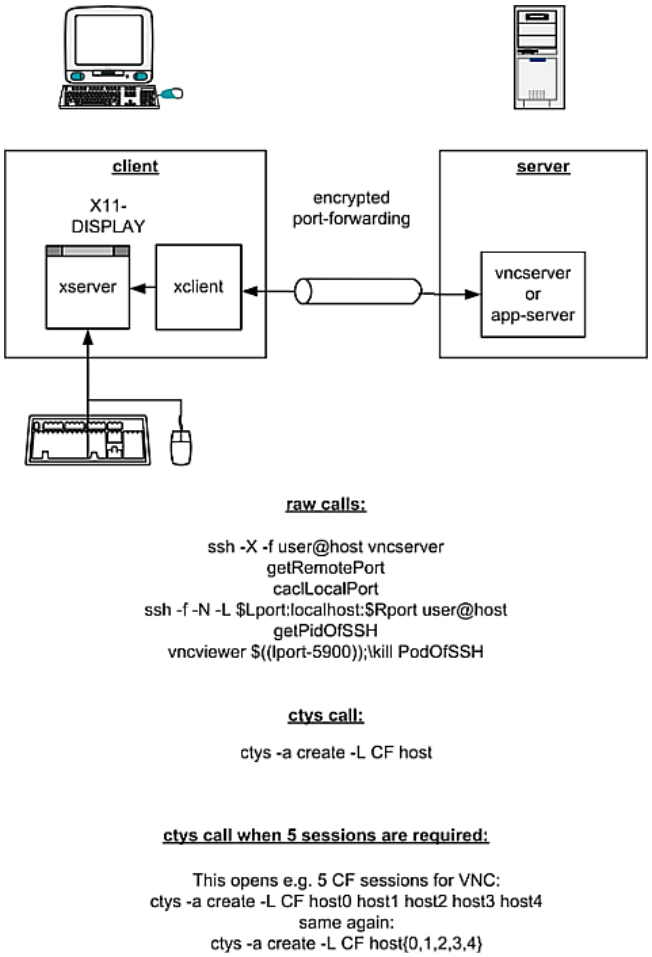

Figure 14: DISPLAYFORWARDING

The instances could be listed by the LIST action in several variants. The basic call with default selection executed on the caller workstation is:

```
ctys -a list ws2
```

The standard assignment to LIST call is "tab_tcp,both", which displays:

```
TCP-container|TCP-guest|label |sesstype|c|user|group
-------------+---------+------+--------+-+----+---------
ws2.soho     |-        |tst000|VNC     |C|acue|ldapusers
ws2.soho     |-        |tst001|VNC     |C|acue|ldapusers
ws2.soho     |ws2.soho.|ws2   |PM      |S|-   |-
ws2.soho     |-        |tst000|SSH(VNC)|T|acue|ldapusers
ws2.soho     |-        |tst001|SSH(VNC)|T|acue|ldapusers
ws2.soho     |-        |tst000|VNC     |C|acue|ldapusers
ws2.soho     |-        |tst001|VNC     |C|acue|ldapusers
```

Here the two tunnels could be identified as "sesstype=SSH(VNC)", and "c=T". This indicates, that the tunnels are created for the subsystem VNC with the session label "tst000" and "tst001".
The following call displays the same table, but with IDs instead of LABELs.

```
ctys -a list=tab\_tcp,id ws2
```

Which results to the display:

```
TCP-cont|TCP-guest|id        |sesstype|c|user|group
--------+---------+----------+--------+-+----+---------
ws2.soho|-        |50        |VNC     |C|acue|ldapusers
ws2.soho|-        |51        |VNC     |C|acue|ldapusers
ws2.soho|-        |../pm.conf|PM      |S|-   |-
ws2.soho|-        |5950-5903 |SSH(VNC)|T|acue|ldapusers
ws2.soho|-        |5951-5904 |SSH(VNC)|T|acue|ldapusers
ws2.soho|-        |50        |VNC     |C|acue|ldapusers
ws2.soho|-        |51        |VNC     |C|acue|ldapusers
```

Indicating by the default ID of tunnels, that these are tunnels forwarding the ports "5950" to "5903" and "5951" to "5904".
The display could be changed as required by usage of specific free-customized tables, e.g. displaying LABEL and ID columns once.
The call with the whole set of involved machines as one call results to:

```
ctys -a list=tab\_tcp,id ws2 lab00 lab01
```

```
ctys -a list=tab\_tcp,id ws2 lab00 lab01

TCP-contai|TCP-guest|id        |sesstype|c|user|group
----------+---------+----------+--------+-+----+---------
ws2.soho  |-        |50        |VNC     |C|acue|ldapusers
ws2.soho  |-        |51        |VNC     |C|acue|ldapusers
ws2.soho  |-        |d/pm.conf |PM      |S|-   |-
ws2.soho  |-        |5950-5903 |SSH(VNC)|T|acue|ldapusers
ws2.soho  |-        |5951-5904 |SSH(VNC)|T|acue|ldapusers
lab00.soho|-        |3784      |CLI     |C|acue|ldapusers
lab00.soho|-        |31206     |CLI     |C|acue|ldapusers
lab00.soho|-        |1         |VNC     |S|root|root
lab00.soho|-        |2         |VNC     |S|acue|ldapusers
lab00.soho|-        |          |XEN     |S|-   |-
lab00.soho|-        |e/xen/tst1|XEN     |S|-   |-
lab00.soho|-        |d/pm.conf |PM      |S|-   |-
lab01.soho|-        |          |XEN     |S|-   |-
lab01.soho|-        |d/pm.conf |PM      |S|-   |-
```

6 SEE ALSO

ctys(1) , *ctys-groups(1)* , *ctys-macros(1)* , *ctys-plugins(1)* , *ctys-vhost(1)* , *ctys-VNC(1)* , *vncpasswd(1)*, *vncviewer(1)*, *vncserver(1)*
For System Tools:
RealVNC: [http://www.realvnc.com]
TigerVNC: [http://www.tigervnc.org]
TightVNC: [http://www.tightvnc.com]

7 AUTHOR

Maintenance:	<acue_sf1@sourceforge.net>
Homepage:	<http://www.UnifiedSessionsManager.org>
Sourceforge.net:	<http://sourceforge.net/projects/ctys>
Berlios.de:	<http://ctys.berlios.de>
Commercial:	<http://www.i4p.com>

8 COPYRIGHT