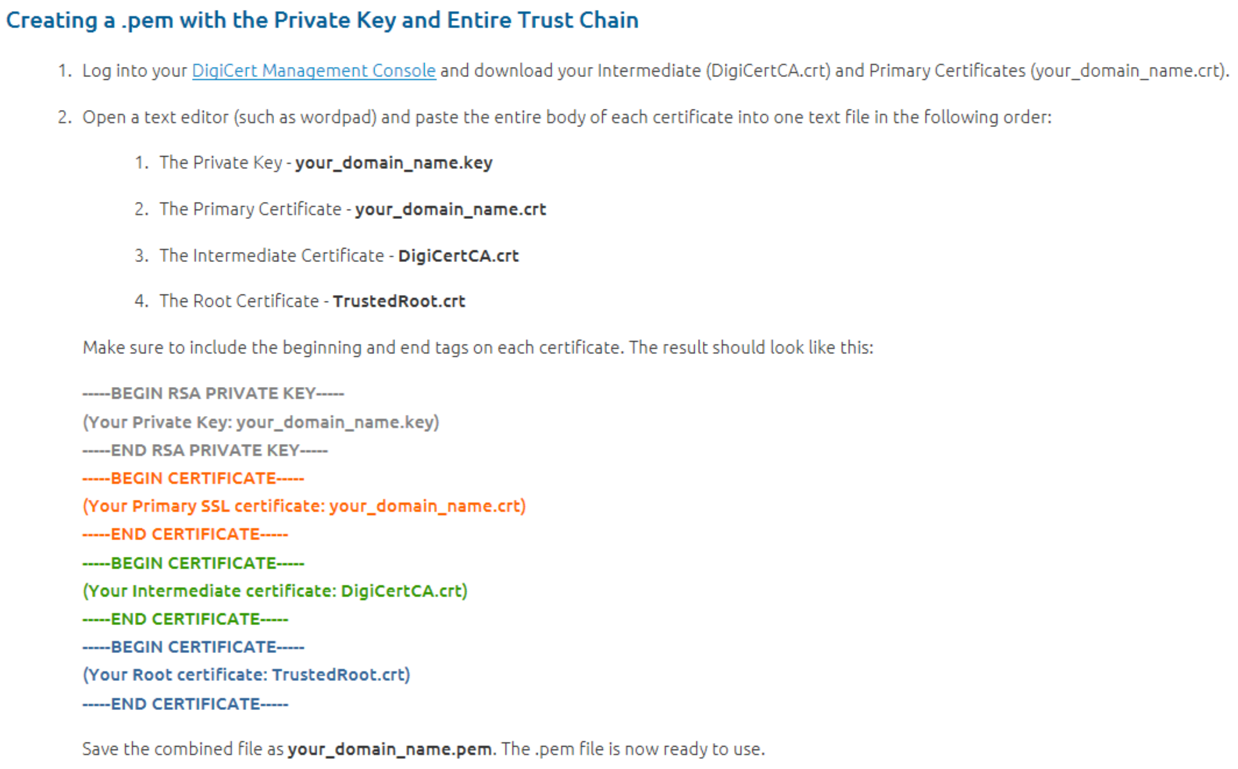
Contents

Imported wild card pem file

Reference: https://www.digicert.com/ssl-support/pem-ssl-creation.htm

1. There are a number of ways to obtain the certificate chain. Probably the best way is to get it from the CA like GeoTrust or Verisign. For these instructions I already had the certificate chain in a PFX file.
2. My PFX file had a password protected key and when I initially converted the PFX to PEM I added a pass phrase. So I had a pass phrase for the key as well as the PEM file. I discovered that UCSD's VNC/websock didn't like the pass phrases so the following procedures are what I needed to do to resolve.
3. Launch MMC and add Certificate snap in.
4. Import exported3.pfx into (local computer) → personal/Certificates folder
5. Find your_domain_name and export it in base64 as primary.cer
6. Find the intermediate and root certificates and export in base64 as intermediate.cer and root.cer
7. To strip the pass phrase from private key. Run the following command
    a. Convert pfx file to pem → `openssl pkcs12 -in filename.pfx -out site.pem`
    b. Strip pass phrase from key →
    ```
    openssl.exe rsa -in ucsd.pem -out
    keynophrase.key
    Enter pass phrase for ucsd.pem:
    writing RSA key
    ```
8. Open keynophrase.key, primary.cer, intermediate.cer and root.cer in notepad and copy/paste all content of each into a new ucsd2.pem file.

(Screen shot is for reference from the digicert.com website)



Creating a .pem with the Private Key and Entire Trust Chain

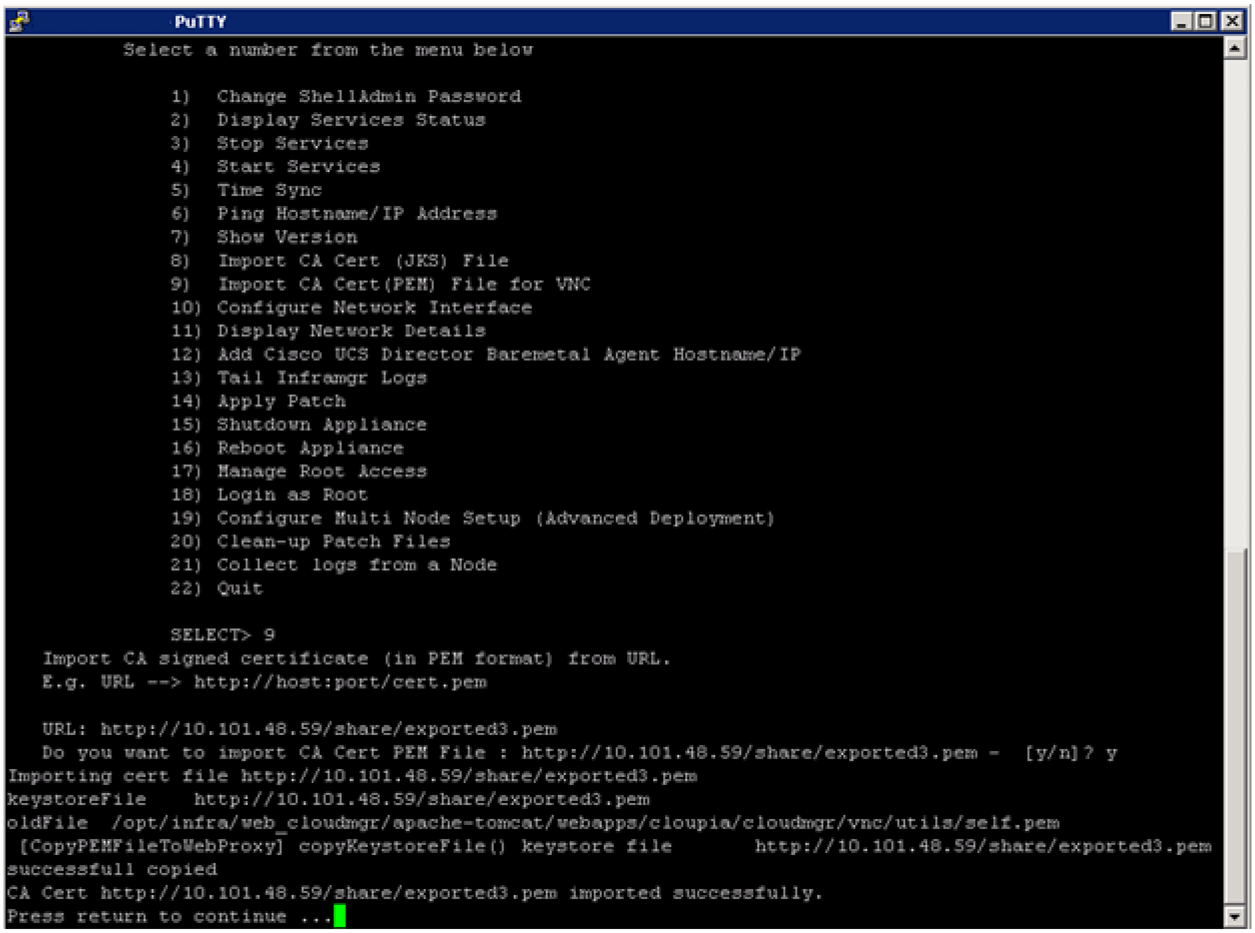
1. Log into your DigiCert Management Console and download your Intermediate (DigiCertCA.crt) and Primary Certificates (your_domain_name.crt).
2. Open a text editor (such as wordpad) and paste the entire body of each certificate into one text file in the following order:
    1. The Private Key - **your_domain_name.key**
    2. The Primary Certificate - **your_domain_name.crt**
    3. The Intermediate Certificate - **DigiCertCA.crt**
    4. The Root Certificate - **TrustedRoot.crt**

    Make sure to include the beginning and end tags on each certificate. The result should look like this:

    ```
    -----BEGIN RSA PRIVATE KEY-----
    (Your Private Key: your_domain_name.key)
    -----END RSA PRIVATE KEY-----
    -----BEGIN CERTIFICATE-----
    (Your Primary SSL certificate: your_domain_name.crt)
    -----END CERTIFICATE-----
    -----BEGIN CERTIFICATE-----
    (Your Intermediate certificate: DigiCertCA.crt)
    -----END CERTIFICATE-----
    -----BEGIN CERTIFICATE-----
    (Your Root certificate: TrustedRoot.crt)
    -----END CERTIFICATE-----
    ```

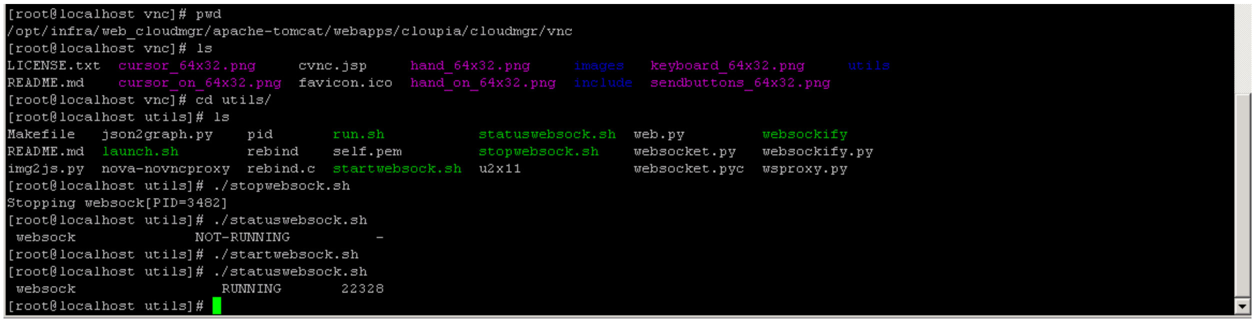
    Save the combined file as **your_domain_name.pem**. The .pem file is now ready to use.

9. Copied ucsd2.pem to http://webserver/share/exported3.pem, then SSH shelladmin into ucsd primary, selected option 9 to import CA Cert (PEM) file for VNC



```
        Select a number from the menu below

            1)  Change ShellAdmin Password
            2)  Display Services Status
            3)  Stop Services
            4)  Start Services
            5)  Time Sync
            6)  Ping Hostname/IP Address
            7)  Show Version
            8)  Import CA Cert (JKS) File
            9)  Import CA Cert(PEM) File for VNC
            10) Configure Network Interface
            11) Display Network Details
            12) Add Cisco UCS Director Baremetal Agent Hostname/IP
            13) Tail Inframgr Logs
            14) Apply Patch
            15) Shutdown Appliance
            16) Reboot Appliance
            17) Manage Root Access
            18) Login as Root
            19) Configure Multi Node Setup (Advanced Deployment)
            20) Clean-up Patch Files
            21) Collect logs from a Node
            22) Quit

            SELECT> 9
   Import CA signed certificate (in PEM format) from URL.
   E.g. URL --> http://host:port/cert.pem

   URL: http://10.101.48.59/share/exported3.pem
   Do you want to import CA Cert PEM File : http://10.101.48.59/share/exported3.pem -  [y/n]? y
Importing cert file http://10.101.48.59/share/exported3.pem
keystoreFile    http://10.101.48.59/share/exported3.pem
oldFile  /opt/infra/web_cloudmgr/apache-tomcat/webapps/cloupia/cloudmgr/vnc/utils/self.pem
 [CopyPEMFileToWebProxy] copyKeystoreFile() keystore file       http://10.101.48.59/share/exported3.pem
successfull copied
CA Cert http://10.101.48.59/share/exported3.pem imported successfully.
Press return to continue ...
```

10. Login as root and navigate to /opt/infra/web_cloudmgr/apache-tomcat/webapps/cloupia/cloudmgr/vnc/utils
11. Restarted services: Run ./stopwebsock.sh to stop service and ./startwebsock.sh to start then ./statuswebsock.sh to see if service is running.



```
[root@localhost vnc]# pwd
/opt/infra/web_cloudmgr/apache-tomcat/webapps/cloupia/cloudmgr/vnc
[root@localhost vnc]# ls
LICENSE.txt  cursor_64x32.png     cvnc.jsp     hand_64x32.png     images    keyboard_64x32.png     utils
README.md    cursor_on_64x32.png  favicon.ico  hand_on_64x32.png  include   sendbuttons_64x32.png
[root@localhost vnc]# cd utils/
[root@localhost utils]# ls
Makefile   json2graph.py    pid       run.sh            statuswebsock.sh  web.py         websockify
README.md  launch.sh        rebind    self.pem          stopwebsock.sh    websocket.py   websockify.py
img2js.py  nova-novncproxy  rebind.c  startwebsock.sh   u2x11             websocket.pyc  wsproxy.py
[root@localhost utils]# ./stopwebsock.sh
Stopping websock[PID=3482]
[root@localhost utils]# ./statuswebsock.sh
 websock              NOT-RUNNING          -
[root@localhost utils]# ./startwebsock.sh
[root@localhost utils]# ./statuswebsock.sh
 websock                 RUNNING       22328
[root@localhost utils]#
```

External Firewall Ports

You need 443 and 8787 open to UCSD

UCSD needs 5900 – 5964 open to the ESXi hosts.

VIB and ESXi Host Configurations for Persistent Firewall Rule:

Reference: http://www.yellow-bricks.com/2011/11/29/how-to-create-your-own-vib-files/

1. This is not supported by vmware but you will need to keep VNC ports open.
2. Created a RHEL 7 vm
3. Winscp copied E:\Cisco\Nexus1000v\Cisco_bootbank_cisco-vem-v172-esx_5.2.1.3.1.3.0-3.2.1.vib to the VM
4. SSH'd into VM. Ran more Cisco_bootbank_cisco-vem-v172-esx_5.2.1.3.1.3.0-3.2.1.vib to confirm it was full of binaries.
5. Ran ar tv Cisco_bootbank_cisco-vem-v172-esx_5.2.1.3.1.3.0-3.2.1.vib which output contents of file.

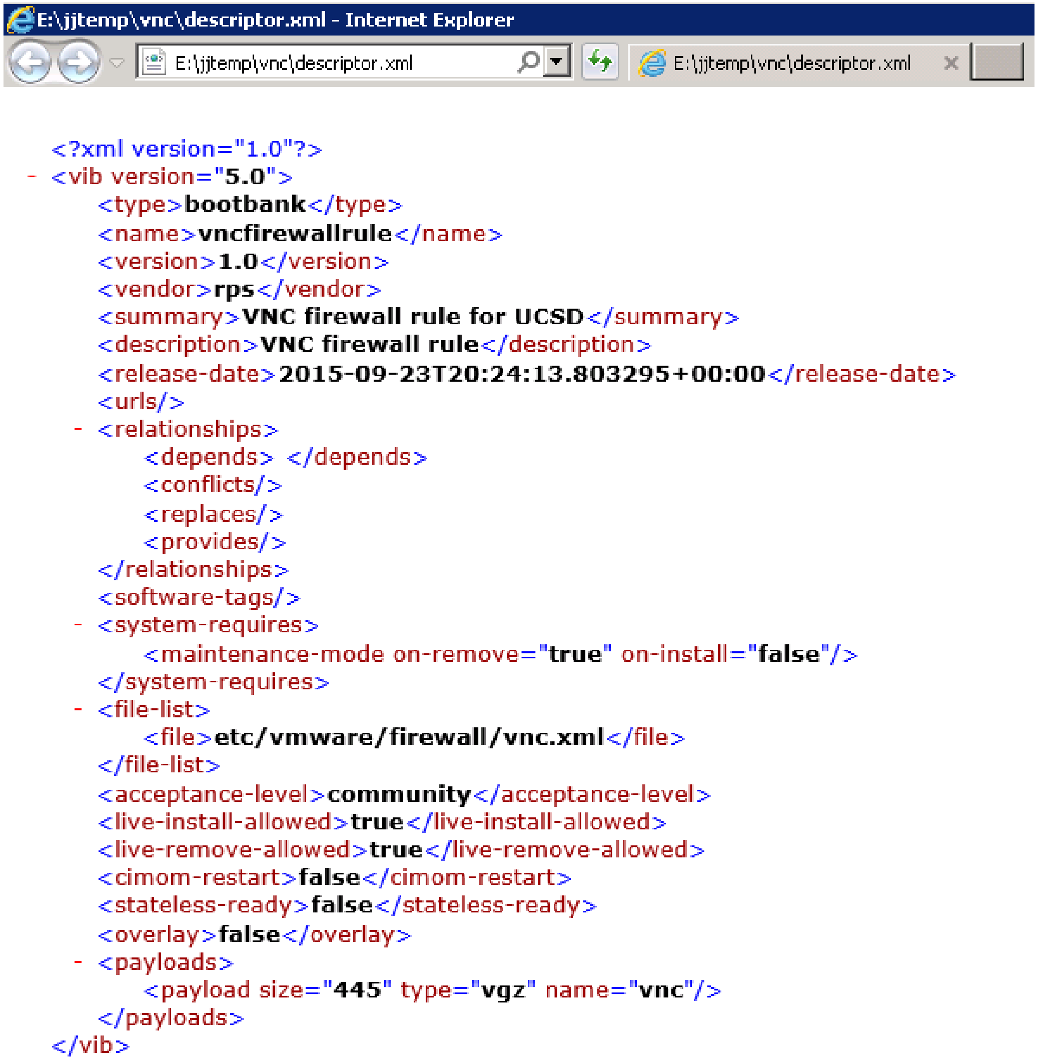
a. [root@localhost vibauth]# ar tv Cisco_bootbank_cisco-vem-v172-esx_5.2.1.3.1.3.0-3.2.1.vib

```
--------- 0/0   8171 Dec 31 19:00 1969 descriptor.xml
--------- 0/0   2090 Dec 31 19:00 1969 sig.pkcs7
--------- 0/0 7324508 Dec 31 19:00 1969 cisco-vem-v172-
```

6. Run tar –tzvf cisco-vem-v172- to show contents
7. Run tar –xzvf cisco-vem-v172- to extract the contents
8. Added firewall folder under /etc/vmware
9. Copied vnc.xml into the firewall folder
10. Then package the directories and file
    a. [root@localhost vibauth]# tar -czvf vnc etc/

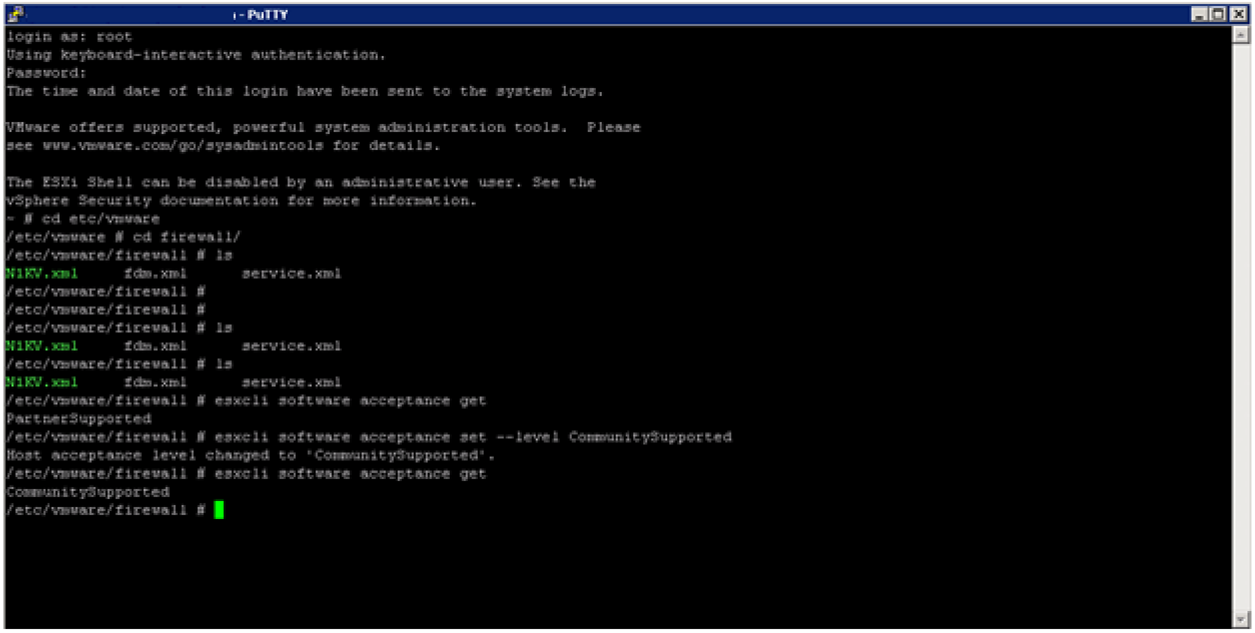
```
etc/
etc/vmware/
etc/vmware/firewall/
etc/vmware/firewall/vnc.xml
```

11. Edit descriptor.xml to the following


E:\jjtemp\vnc\descriptor.xml - Internet Explorer
E:\jjtemp\vnc\descriptor.xml
E:\jjtemp\vnc\descriptor.xml

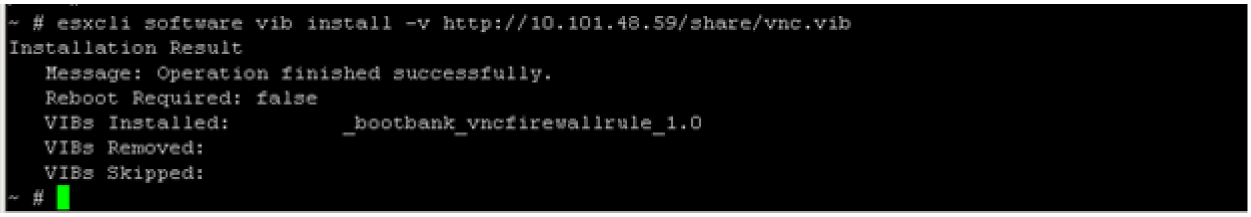
```
  <?xml version="1.0"?>
- <vib version="5.0">
      <type>bootbank</type>
      <name>vncfirewallrule</name>
      <version>1.0</version>
      <vendor>rps</vendor>
      <summary>VNC firewall rule for UCSD</summary>
      <description>VNC firewall rule</description>
      <release-date>2015-09-23T20:24:13.803295+00:00</release-date>
      <urls/>
    - <relationships>
          <depends> </depends>
          <conflicts/>
          <replaces/>
          <provides/>
      </relationships>
      <software-tags/>
    - <system-requires>
          <maintenance-mode on-remove="true" on-install="false"/>
      </system-requires>
    - <file-list>
          <file>etc/vmware/firewall/vnc.xml</file>
      </file-list>
      <acceptance-level>community</acceptance-level>
      <live-install-allowed>true</live-install-allowed>
      <live-remove-allowed>true</live-remove-allowed>
      <cimom-restart>false</cimom-restart>
      <stateless-ready>false</stateless-ready>
      <overlay>false</overlay>
    - <payloads>
          <payload size="445" type="vgz" name="vnc"/>
      </payloads>
  </vib>
```

12. Make the new VIB. Run ar -r vnc.vib descriptor.xml sig.pkcs7 vnc (note the order of the files, this is the order esxi needs to correctly install)
13. SSH into ESXi host and set software acceptance level to CommunitySupported.



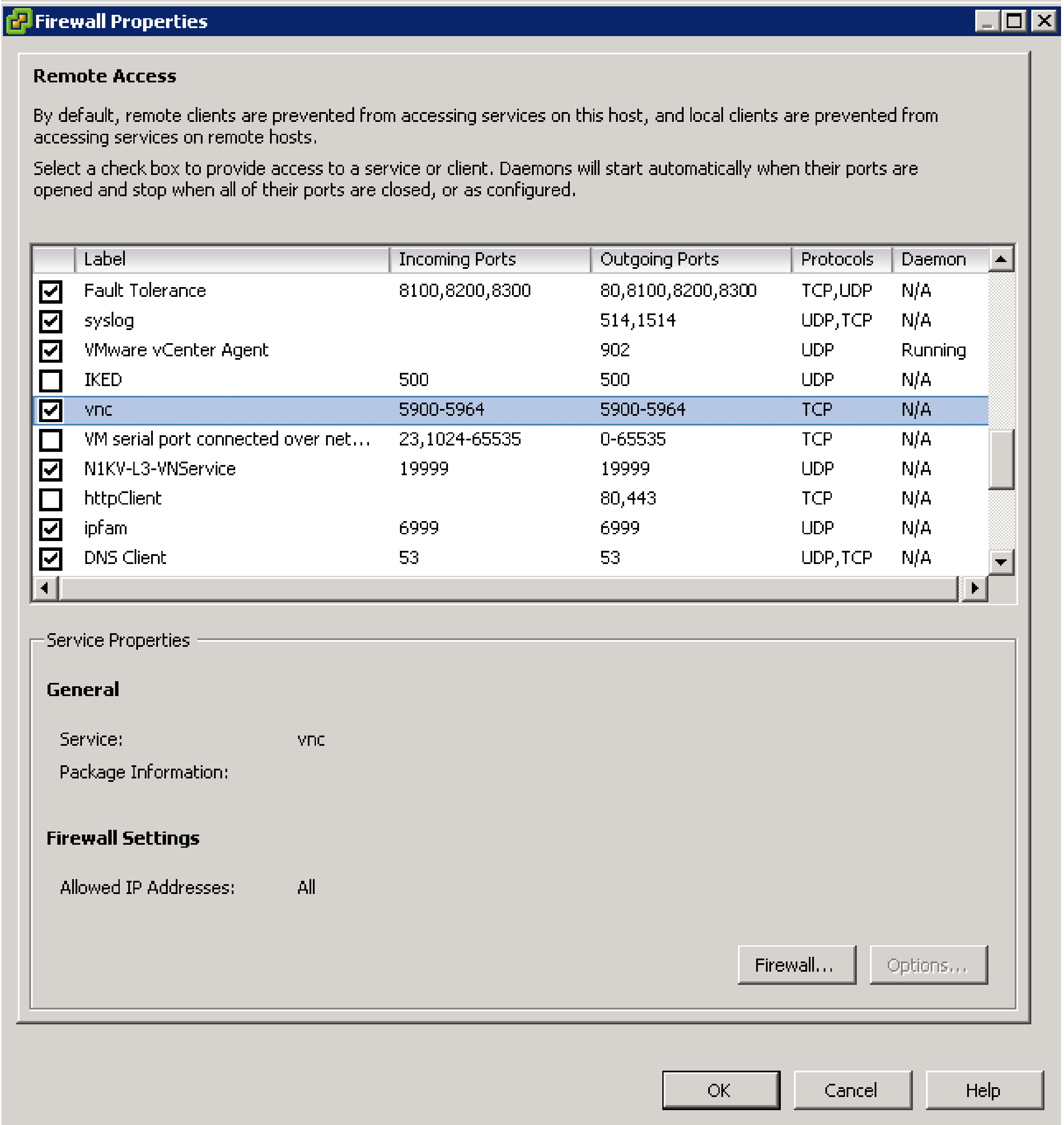
```
login as: root
Using keyboard-interactive authentication.
Password:
The time and date of this login have been sent to the system logs.

VMware offers supported, powerful system administration tools.  Please
see www.vmware.com/go/sysadmintools for details.

The ESXi Shell can be disabled by an administrative user. See the
vSphere Security documentation for more information.
~ # cd etc/vmware
/etc/vmware # cd firewall/
/etc/vmware/firewall # ls
N1KV.xml     fdm.xml      service.xml
/etc/vmware/firewall #
/etc/vmware/firewall #
/etc/vmware/firewall # ls
N1KV.xml     fdm.xml      service.xml
/etc/vmware/firewall # ls
N1KV.xml     fdm.xml      service.xml
/etc/vmware/firewall # esxcli software acceptance get
PartnerSupported
/etc/vmware/firewall # esxcli software acceptance set --level CommunitySupported
Host acceptance level changed to 'CommunitySupported'.
/etc/vmware/firewall # esxcli software acceptance get
CommunitySupported
/etc/vmware/firewall #
```

14. Copied the vnc.vib file up to my web server and ran the install command below from ESXi
    esxcli software vib install –v http://10.101.48.59/share/vnc.vib



```
~ # esxcli software vib install -v http://10.101.48.59/share/vnc.vib
Installation Result
   Message: Operation finished successfully.
   Reboot Required: false
   VIBs Installed:          _bootbank_vncfirewallrule_1.0
   VIBs Removed:
   VIBs Skipped:
~ #
```

Refresh security profile → firewall and confirm VNC



Remote Access

By default, remote clients are prevented from accessing services on this host, and local clients are prevented from accessing services on remote hosts.

Select a check box to provide access to a service or client. Daemons will start automatically when their ports are opened and stop when all of their ports are closed, or as configured.

	Label	Incoming Ports	Outgoing Ports	Protocols	Daemon
☑	Fault Tolerance	8100,8200,8300	80,8100,8200,8300	TCP,UDP	N/A
☑	syslog		514,1514	UDP,TCP	N/A
☑	VMware vCenter Agent		902	UDP	Running
□	IKED	500	500	UDP	N/A
☑	vnc	5900-5964	5900-5964	TCP	N/A
□	VM serial port connected over net...	23,1024-65535	0-65535	TCP	N/A
☑	N1KV-L3-VNService	19999	19999	UDP	N/A
□	httpClient		80,443	TCP	N/A
☑	ipfam	6999	6999	UDP	N/A
☑	DNS Client	53	53	UDP,TCP	N/A

Service Properties

General

Service: vnc

Package Information:

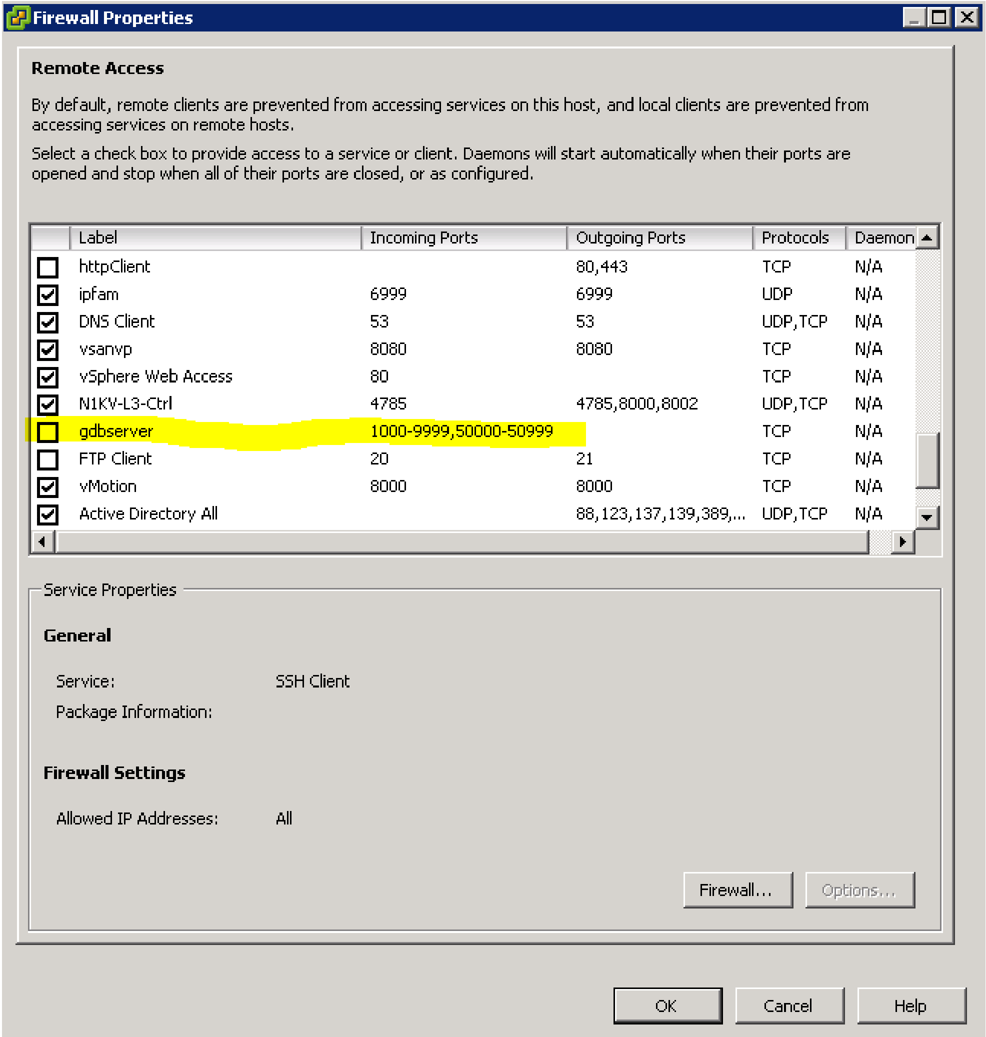
Firewall Settings

Allowed IP Addresses: All

15. Rebooted host to confirm persistence.

Alternative ESXi host firewall rule configuration (probably better because it would be supported by VMware)

1. Go to your ESXi host select Configuration tab →Security Profile → scroll down to Firewall and select Properties
2. Scroll down to gdbserver and check the box to allow incoming ports



Firewall Properties

Remote Access

By default, remote clients are prevented from accessing services on this host, and local clients are prevented from accessing services on remote hosts.

Select a check box to provide access to a service or client. Daemons will start automatically when their ports are opened and stop when all of their ports are closed, or as configured.

	Label	Incoming Ports	Outgoing Ports	Protocols	Daemon
□	httpClient		80,443	TCP	N/A
☑	ipfam	6999	6999	UDP	N/A
☑	DNS Client	53	53	UDP,TCP	N/A
☑	vsanvp	8080	8080	TCP	N/A
☑	vSphere Web Access	80		TCP	N/A
☑	N1KV-L3-Ctrl	4785	4785,8000,8002	UDP,TCP	N/A
□	gdbserver	1000-9999,50000-50999		TCP	N/A
□	FTP Client	20	21	TCP	N/A
☑	vMotion	8000	8000	TCP	N/A
☑	Active Directory All		88,123,137,139,389,...	UDP,TCP	N/A

Service Properties

General

Service: SSH Client

Package Information:

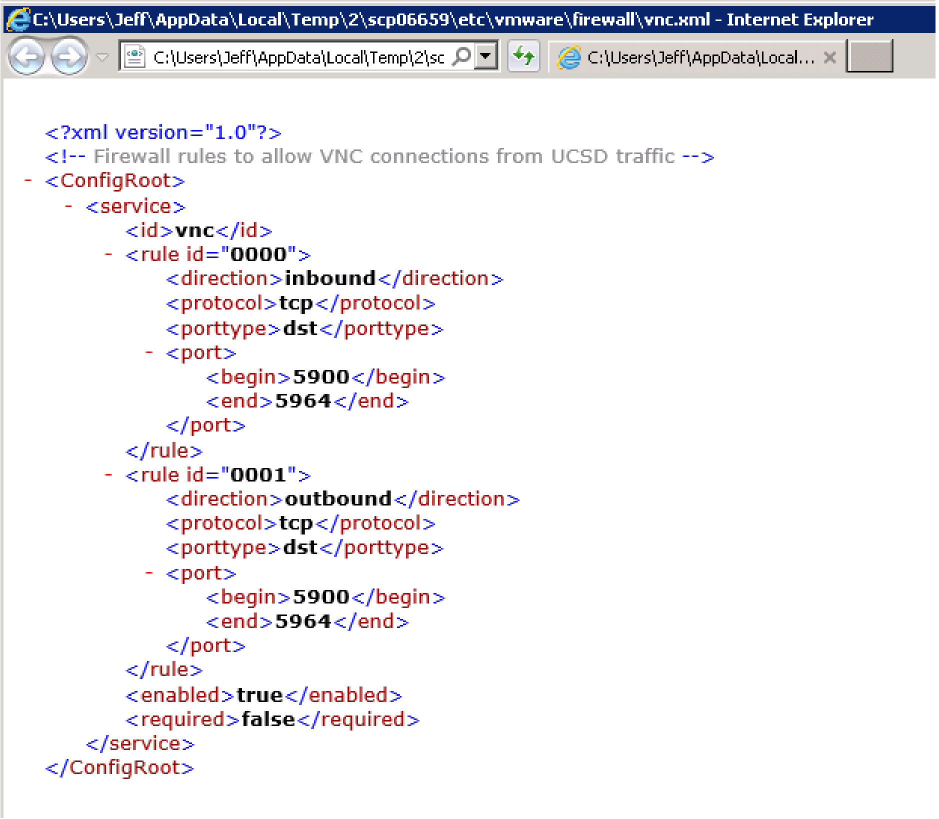
Firewall Settings

Allowed IP Addresses: All

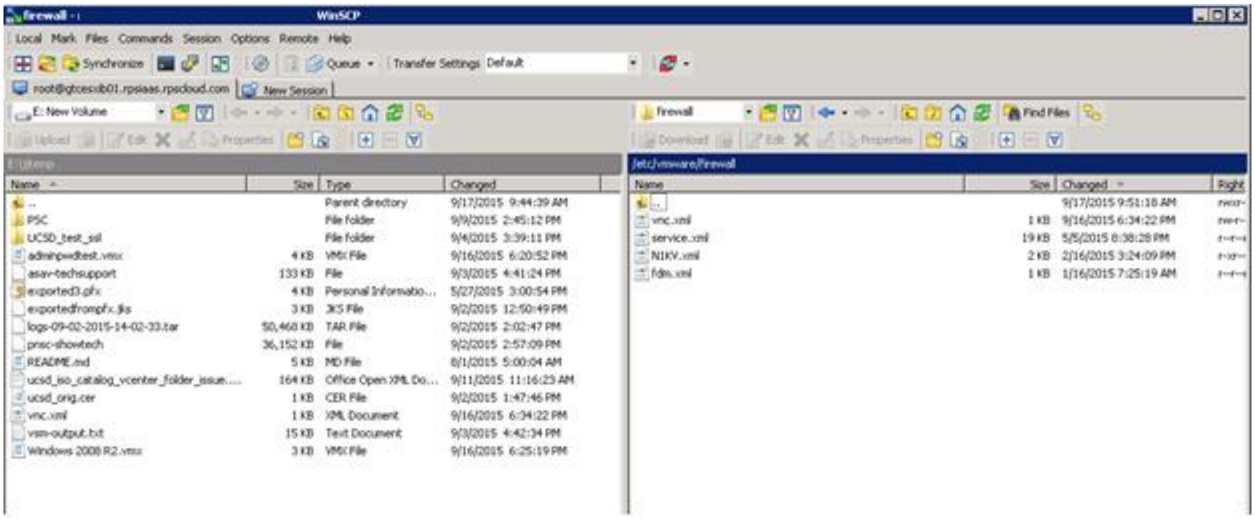
3. You should have a firewall between UCSD and your ESXi hosts so you can restrict ports to 5900 – 5964.

Configure ESXi for test VNC Firewall Rule (non-persistent)

1. Created vnc.xml and copied to /etc/vmware/firewall on ESXi host


C:\Users\Jeff\AppData\Local\Temp\2\scp06659\etc\vmware\firewall\vnc.xml - Internet Explorer
C:\Users\Jeff\AppData\Local\Temp\2\sc
C:\Users\Jeff\AppData\Local...

```
  <?xml version="1.0"?>
  <!-- Firewall rules to allow VNC connections from UCSD traffic -->
- <ConfigRoot>
  - <service>
      <id>vnc</id>
    - <rule id="0000">
        <direction>inbound</direction>
        <protocol>tcp</protocol>
        <porttype>dst</porttype>
      - <port>
          <begin>5900</begin>
          <end>5964</end>
        </port>
      </rule>
    - <rule id="0001">
        <direction>outbound</direction>
        <protocol>tcp</protocol>
        <porttype>dst</porttype>
      - <port>
          <begin>5900</begin>
          <end>5964</end>
        </port>
      </rule>
      <enabled>true</enabled>
      <required>false</required>
    </service>
</ConfigRoot>
```



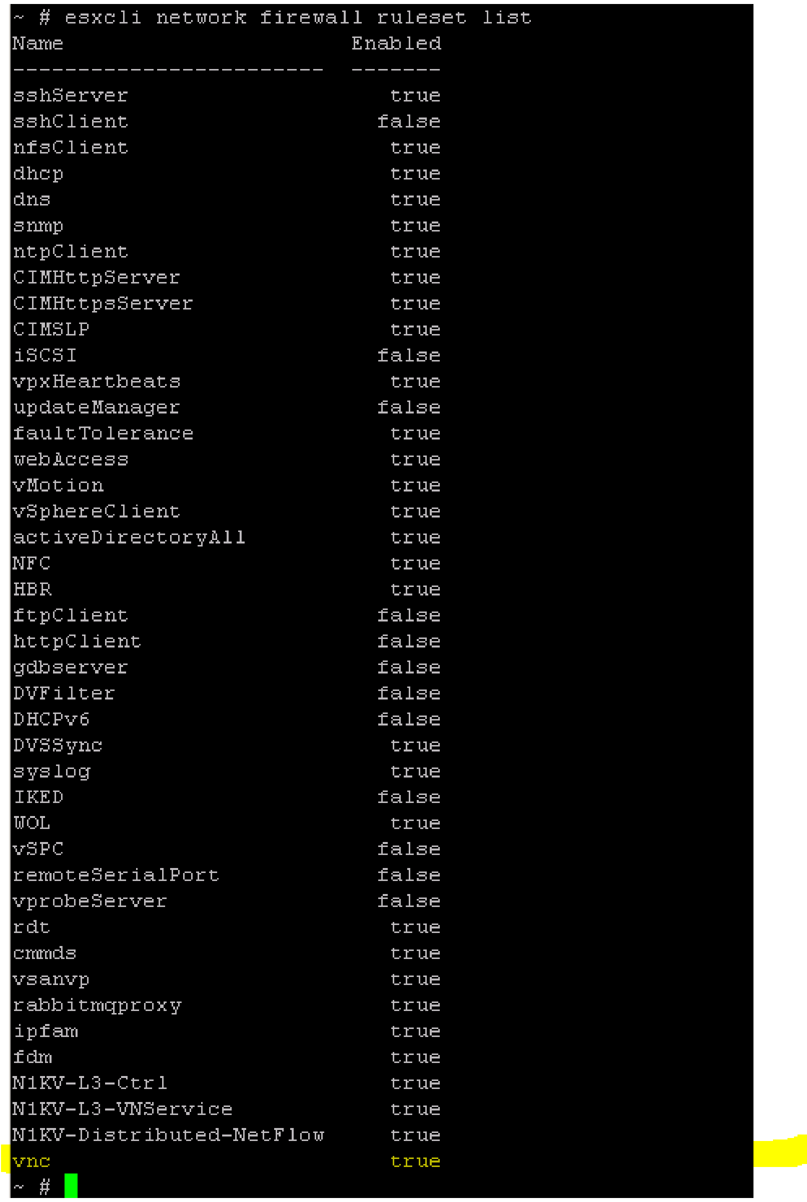
Name	Size	Type	Changed
..		Parent directory	9/17/2015 9:44:39 AM
PSC		File folder	9/9/2015 2:45:12 PM
UCSD_test_ssl		File folder	9/4/2015 3:39:11 PM
adminpwdtest.vmx	4 KB	VMX File	9/16/2015 6:20:52 PM
asav-techsupport	133 KB	File	9/3/2015 4:41:24 PM
exported3.pfx	4 KB	Personal Informatio...	5/27/2015 3:00:54 PM
exportedfrompfx.jks	3 KB	JKS File	9/2/2015 12:50:49 PM
logs-09-02-2015-14-02-33.tar	50,460 KB	TAR File	9/2/2015 2:02:47 PM
pnsc-showtech	36,152 KB	File	9/2/2015 2:57:09 PM
README.md	5 KB	MD File	8/1/2015 5:00:04 AM
ucsd_iso_catalog_vcenter_folder_issue....	164 KB	Office Open XML Do...	9/11/2015 11:16:23 AM
ucsd_orig.cer	1 KB	CER File	9/2/2015 1:47:46 PM
vnc.xml	1 KB	XML Document	9/16/2015 6:34:22 PM
vsm-output.txt	15 KB	Text Document	9/3/2015 4:42:34 PM
Windows 2008 R2.vmx	3 KB	VMX File	9/16/2015 6:25:19 PM

Name	Size	Changed	Right
..		9/17/2015 9:51:18 AM	rwxr-
vnc.xml	1 KB	9/16/2015 6:34:22 PM	rw-r--
service.xml	19 KB	5/5/2015 8:38:28 PM	r--r--r
N1KV.xml	2 KB	2/16/2015 3:24:09 PM	r-xr--
fdm.xml	1 KB	1/16/2015 7:25:19 AM	r--r--r

2. Confirmed .vmx configuration on test vm for vnc settings.

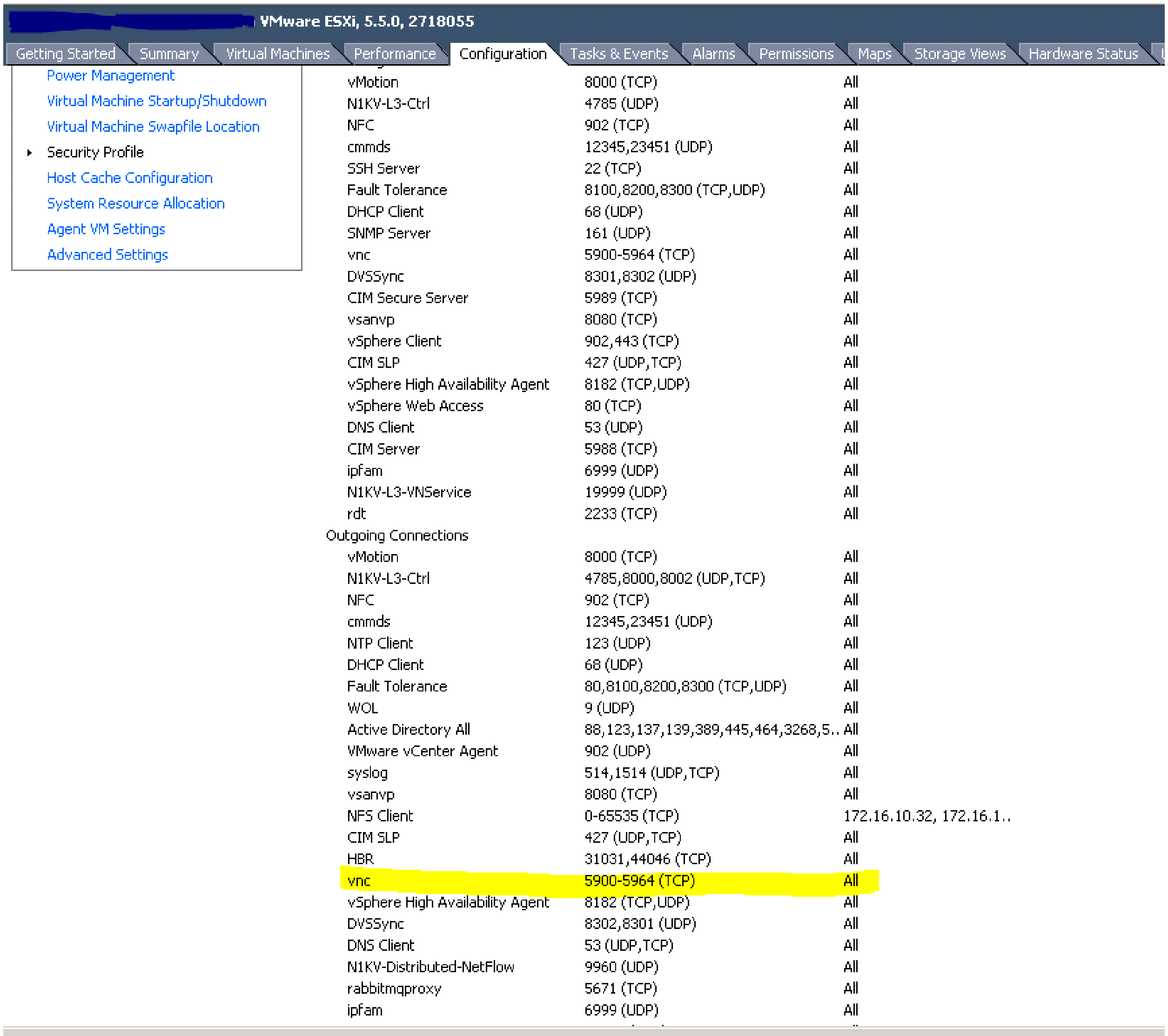
```
RemoteDisplay.vnc.enabled = "TRUE"
RemoteDisplay.vnc.port = "5957"
RemoteDisplay.vnc.password = "abc123"
RemoteDisplay.vnc.key =
```

3. SSH into esxi host and run esxcli network firewall ruleset list to list current firewall rules
4. Run esxcli network firewall refresh to update rules



```
~ # esxcli network firewall ruleset list
Name                      Enabled
------------------------  -------
sshServer                    true
sshClient                   false
nfsClient                    true
dhcp                         true
dns                          true
snmp                         true
ntpClient                    true
CIMHttpServer                true
CIMHttpsServer               true
CIMSLP                       true
iSCSI                       false
vpxHeartbeats                true
updateManager               false
faultTolerance               true
webAccess                    true
vMotion                      true
vSphereClient                true
activeDirectoryAll           true
NFC                          true
HBR                          true
ftpClient                   false
httpClient                  false
gdbserver                   false
DVFilter                    false
DHCPv6                      false
DVSSync                      true
syslog                       true
IKED                        false
WOL                          true
vSPC                        false
remoteSerialPort            false
vprobeServer                false
rdt                          true
cmmds                        true
vsanvp                       true
rabbitmqproxy                true
ipfam                        true
fdm                          true
N1KV-L3-Ctrl                 true
N1KV-L3-VNService            true
N1KV-Distributed-NetFlow     true
vnc                          true
~ #
```

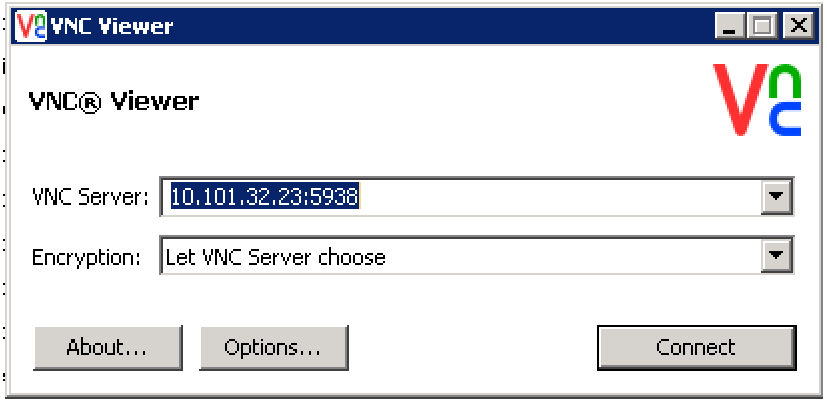
5. From vsphere web client check new rule exists.



[redacted] VMware ESXi, 5.5.0, 2718055

vMotion	8000 (TCP)	All
N1KV-L3-Ctrl	4785 (UDP)	All
NFC	902 (TCP)	All
cmmds	12345,23451 (UDP)	All
SSH Server	22 (TCP)	All
Fault Tolerance	8100,8200,8300 (TCP,UDP)	All
DHCP Client	68 (UDP)	All
SNMP Server	161 (UDP)	All
vnc	5900-5964 (TCP)	All
DVSSync	8301,8302 (UDP)	All
CIM Secure Server	5989 (TCP)	All
vsanvp	8080 (TCP)	All
vSphere Client	902,443 (TCP)	All
CIM SLP	427 (UDP,TCP)	All
vSphere High Availability Agent	8182 (TCP,UDP)	All
vSphere Web Access	80 (TCP)	All
DNS Client	53 (UDP)	All
CIM Server	5988 (TCP)	All
ipfam	6999 (UDP)	All
N1KV-L3-VNService	19999 (UDP)	All
rdt	2233 (TCP)	All
Outgoing Connections		
vMotion	8000 (TCP)	All
N1KV-L3-Ctrl	4785,8000,8002 (UDP,TCP)	All
NFC	902 (TCP)	All
cmmds	12345,23451 (UDP)	All
NTP Client	123 (UDP)	All
DHCP Client	68 (UDP)	All
Fault Tolerance	80,8100,8200,8300 (TCP,UDP)	All
WOL	9 (UDP)	All
Active Directory All	88,123,137,139,389,445,464,3268,5..	All
VMware vCenter Agent	902 (UDP)	All
syslog	514,1514 (UDP,TCP)	All
vsanvp	8080 (TCP)	All
NFS Client	0-65535 (TCP)	172.16.10.32, 172.16.1..
CIM SLP	427 (UDP,TCP)	All
HBR	31031,44046 (TCP)	All
vnc	5900-5964 (TCP)	All
vSphere High Availability Agent	8182 (TCP,UDP)	All
DVSSync	8302,8301 (UDP)	All
DNS Client	53 (UDP,TCP)	All
N1KV-Distributed-NetFlow	9960 (UDP)	All
rabbitmqproxy	5671 (TCP)	All
ipfam	6999 (UDP)	All

6. Downloaded VNC-Viewer to test connectivity (VNC-Viewer-5.2.3-Windows-64bit.exe) in E:

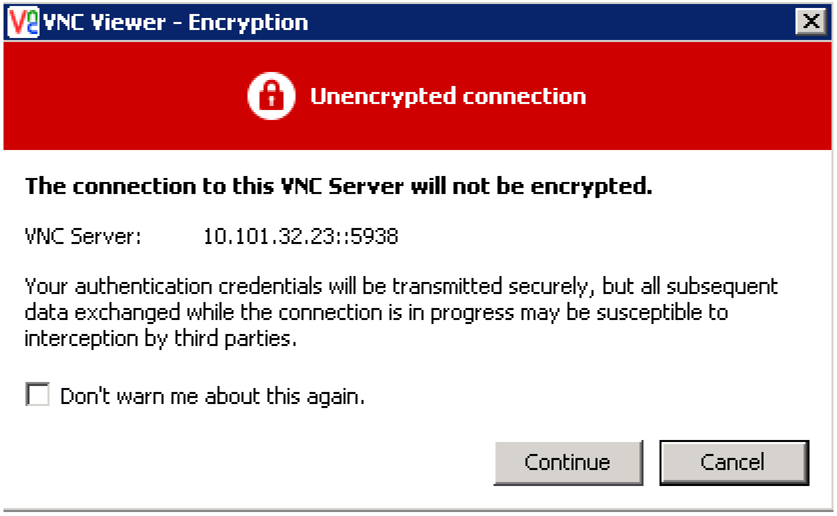


VNC Viewer

VNC® Viewer

VNC Server: 10.101.32.23:5938

Encryption: Let VNC Server choose



VNC Viewer - Encryption

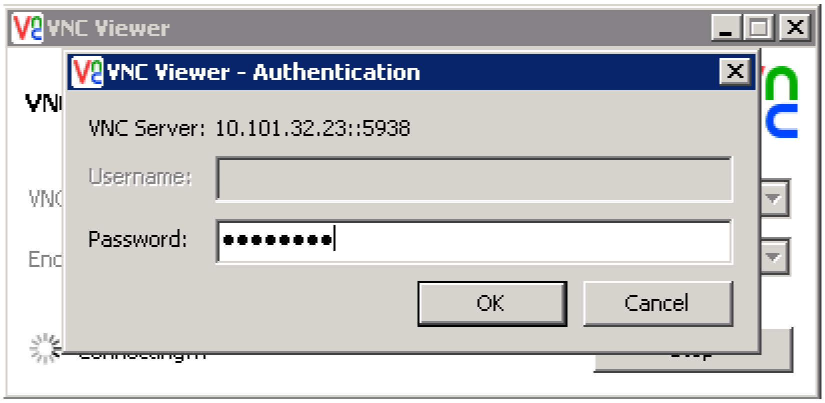
Unencrypted connection

The connection to this VNC Server will not be encrypted.

VNC Server: 10.101.32.23::5938

Your authentication credentials will be transmitted securely, but all subsequent data exchanged while the connection is in progress may be susceptible to interception by third parties.

☐ Don't warn me about this again.



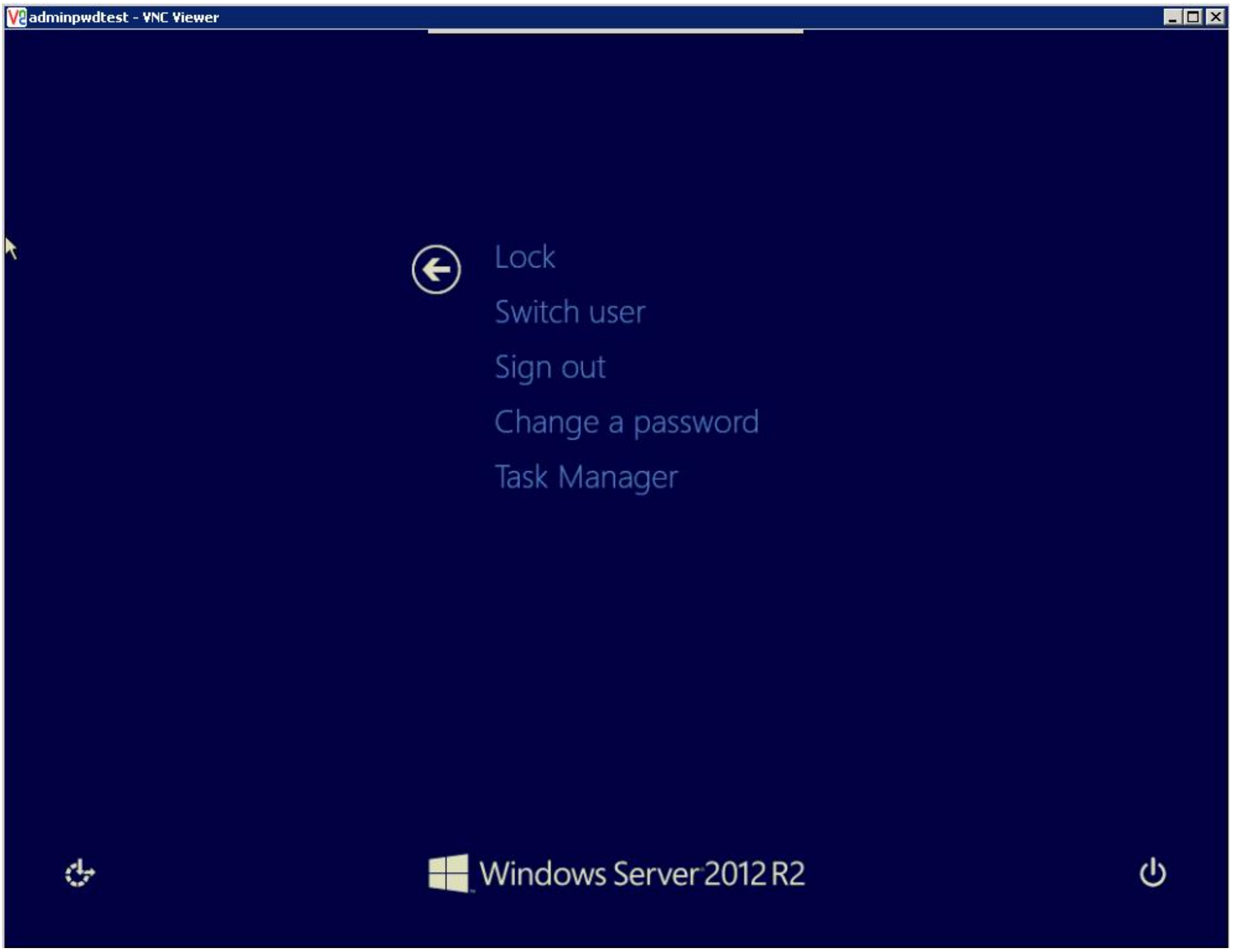
VNC Viewer

VNC Viewer - Authentication

VNC Server: 10.101.32.23::5938

Username:

Password: ●●●●●●●●●

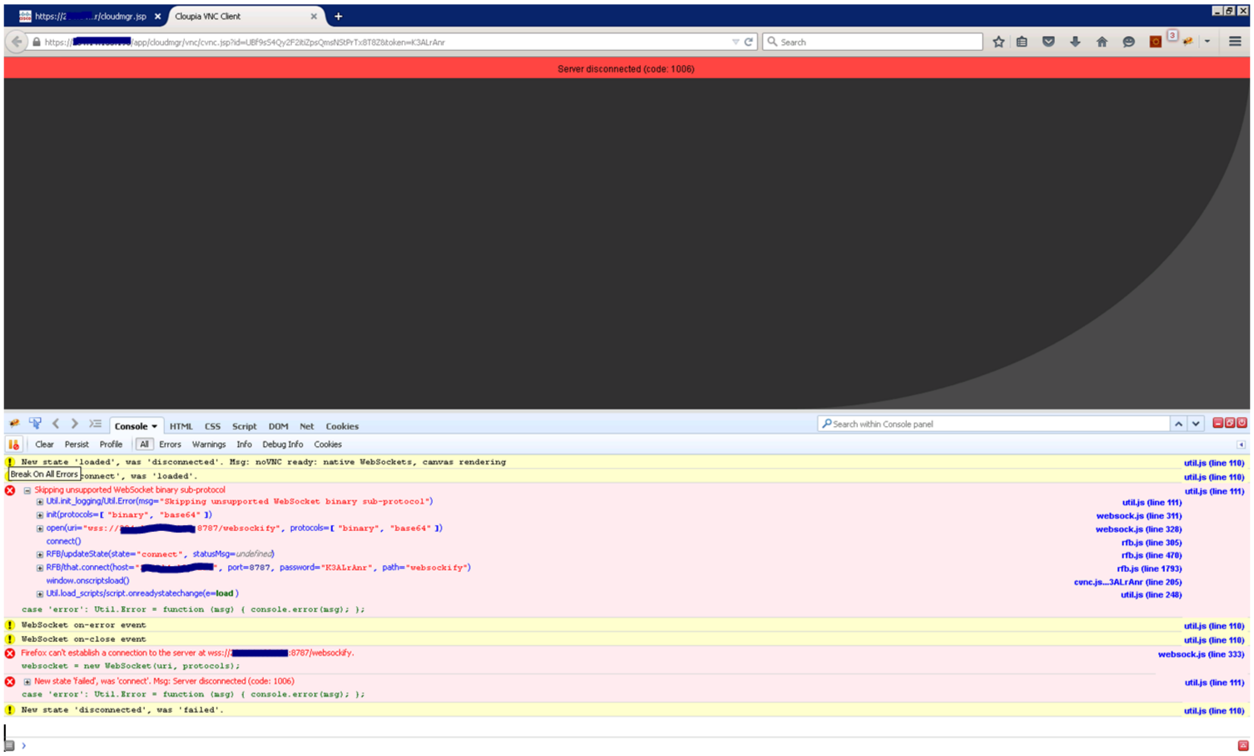


VNC Troubleshooting

noVNC troubleshooting - https://github.com/kanaka/noVNC/wiki/Troubleshooting

Downloaded noVNC (kanaka-noVNC-v0.5.1-81-g40b35fa.zip) and entered host, port and password info and made connection

Used Firebug loaded in Firefox to get debug information.

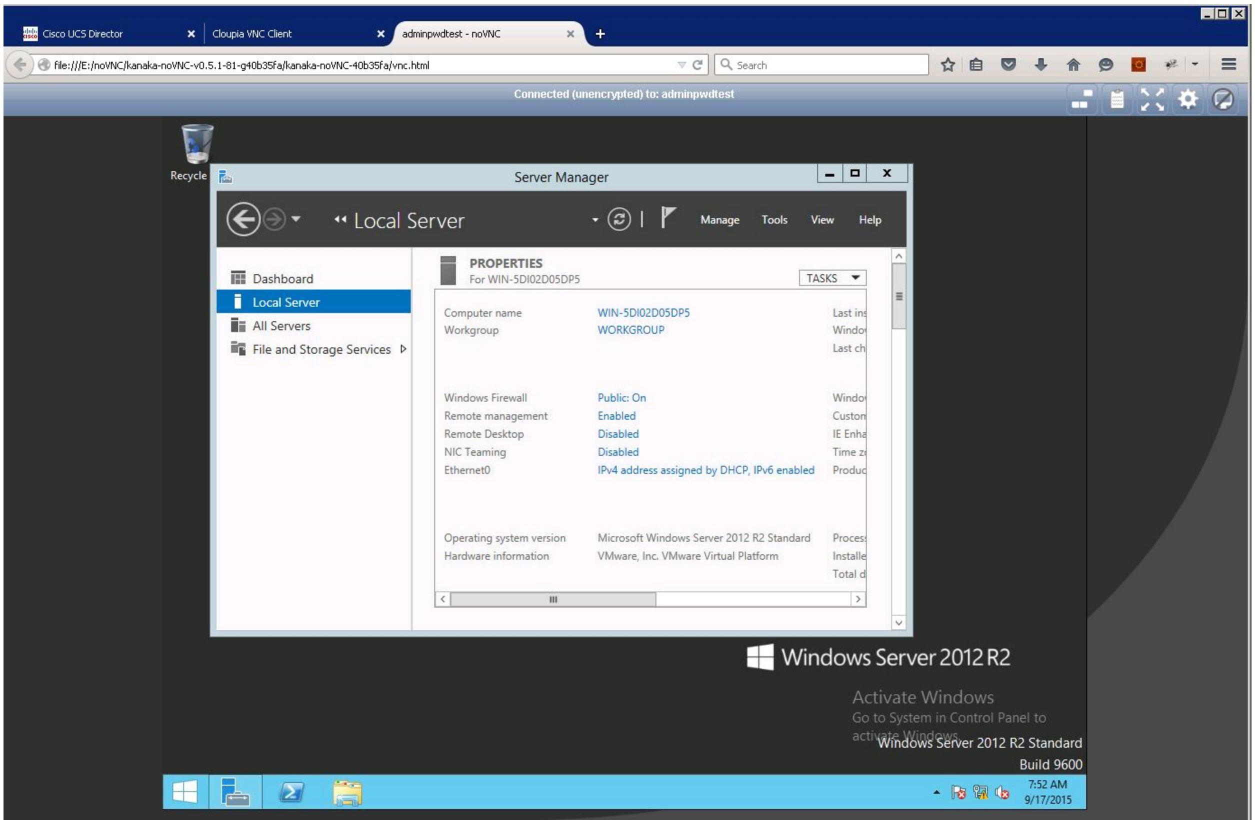


https://[redacted]r/cloudmgr.jsp   Cloupia VNC Client

https://[redacted]/app/cloudmgr/vnc/cvnc.jsp?id=UBf9sS4Qy2F2itiZpsQmsNStPrTx8T8Z8token=K3ALrAnr

Server disconnected (code: 1006)

```
New state 'loaded', was 'disconnected'. Msg: noVNC ready: native WebSockets, canvas rendering    util.js (line 110)
connect', was 'loaded'.                                                                         util.js (line 110)
Skipping unsupported WebSocket binary sub-protocol                                              util.js (line 111)
    Util.init_logging/Util.Error(msg="Skipping unsupported WebSocket binary sub-protocol")       util.js (line 111)
    init(protocols=[ "binary", "base64" ])                                                      websock.js (line 311)
    open(uri="wss://[redacted]8787/websockify", protocols=[ "binary", "base64" ])              websock.js (line 328)
    connect()                                                                                   rfb.js (line 305)
    RFB/updateState(state="connect", statusMsg=undefined)                                       rfb.js (line 470)
    RFB/that.connect(host="[redacted]", port=8787, password="K3ALrAnr", path="websockify")      rfb.js (line 1793)
    window.onscriptsload()                                                                      cvnc.js...3ALrAnr (line 205)
    Util.load_scripts/script.onreadystatechange(e=load )                                        util.js (line 248)
case 'error': Util.Error = function (msg) { console.error(msg); };
WebSocket on-error event                                                                        util.js (line 110)
WebSocket on-close event                                                                        util.js (line 110)
Firefox can't establish a connection to the server at wss://[redacted]:8787/websockify.        websock.js (line 333)
websocket = new WebSocket(uri, protocols);
New state 'failed', was 'connect'. Msg: Server disconnected (code: 1006)                        util.js (line 111)
case 'error': Util.Error = function (msg) { console.error(msg); };
New state 'disconnected', was 'failed'.                                                         util.js (line 110)
```



Cisco UCS Director   Cloupia VNC Client   adminpwdtest - noVNC

file:///E:/noVNC/kanaka-noVNC-v0.5.1-81-g40b35fa/kanaka-noVNC-40b35fa/vnc.html

Connected (unencrypted) to: adminpwdtest

PROPERTIES
For WIN-5DI02D05DP5

Computer name	WIN-5DI02D05DP5	Last ins
Workgroup	WORKGROUP	Windo
		Last ch
Windows Firewall	Public: On	Windo
Remote management	Enabled	Custon
Remote Desktop	Disabled	IE Enha
NIC Teaming	Disabled	Time z
Ethernet0	IPv4 address assigned by DHCP, IPv6 enabled	Produc
Operating system version	Microsoft Windows Server 2012 R2 Standard	Proces
Hardware information	VMware, Inc. VMware Virtual Platform	Installe
		Total d

Windows Server 2012 R2

Activate Windows
Go to System in Control Panel to activate Windows.

Windows Server 2012 R2 Standard
Build 9600

7:52 AM
9/17/2015

VNC path in UCSD is /opt/infra/web_cloudmgr/apache-tomcat/webapps/cloupia/cloudmgr/vnc/utils

Vnc readme.md

```
## noVNC: HTML5 VNC Client


### Description

noVNC is a HTML5 VNC client that runs well in any modern browser
including mobile browsers (iPhone/iPad and Android).

More than 16 companies/projects have integrated noVNC into their
products including [Ganeti Web
Manager](http://code.osuosl.org/projects/ganeti-webmgr),
[OpenStack](http://www.openstack.org), and
[OpenNebula](http://opennebula.org/). See [the Projects and Companies
wiki page](https://github.com/kanaka/noVNC/wiki/ProjectsCompanies-using-noVNC)
for more complete list.

### News/help/contact

Notable commits, announcements and news are posted to
@<a href="http://www.twitter.com/noVNC">noVNC</a>

If you are a noVNC developer/integrator/user (or want to be) please
join the <a
href="https://groups.google.com/forum/?fromgroups#!forum/novnc">noVNC
discussion group</a>

Bugs and feature requests can be submitted via [github
issues](https://github.com/kanaka/noVNC/issues). If you are looking
for a place to start contributing to noVNC, a good place to start
would be the issues that I have marked as
["patchwelcome"](https://github.com/kanaka/noVNC/issues?labels=patchwelcome).

If you want to show appreciation for noVNC you could buy something off
my [Amazon
wishlist](http://www.amazon.com/registry/wishlist/XTXFXK39IA8C/?reveal=unpurchased&sort=priority&
layout=compact) or you could donate to a great non-profits such as: [Compassion
International](http://www.compassion.com/), [SIL](http://www.sil.org),
[Habitat for Humanity](http://www.habitat.org), [Electronic Frontier
Foundation](https://www.eff.org/), [Against Malaria
Foundation](http://www.againstmalaria.com/), [Nothing But
Nets](http://www.nothingbutnets.net/), etc.


### Features

* Supports all modern browsers including mobile (iOS, Android)
* Supported VNC encodings: raw, copyrect, rre, hextile, tight, tightPNG
* WebSocket SSL/TLS encryption (i.e. "wss://") support
* 24-bit true color and 8 bit colour mapped
* Supports desktop resize notification/pseudo-encoding
* Local or remote cursor
* Clipboard copy/paste
* Clipping or scolling modes for large remote screens
* Easy site integration and theming (3 example themes included)
* Licensed under the [MPL 2.0](http://www.mozilla.org/MPL/2.0/)

### Screenshots

Running in Chrome before and after connecting:

<img src="http://kanaka.github.com/noVNC/img/noVNC-5.png" width=400>&nbsp;<img
src="http://kanaka.github.com/noVNC/img/noVNC-7.jpg" width=400>

See more screenshots <a href="http://kanaka.github.com/noVNC/screenshots.html">here</a>.
```

### Browser Requirements

* HTML5 Canvas (with createImageData): Chrome, Firefox 3.6+, iOS
  Safari, Opera 11+, Internet Explorer 9+, etc.

* HTML5 WebSockets: For browsers that do not have builtin
  WebSockets support, the project includes
  <a href="http://github.com/gimite/web-socket-js">web-socket-js</a>,
  a WebSockets emulator using Adobe Flash. iOS 4.2+ has built-in
  WebSocket support.

* Fast Javascript Engine: this is not strictly a requirement, but
  without a fast Javascript engine, noVNC might be painfully slow.

* I maintain a more detailed browser compatibility list <a
  href="https://github.com/kanaka/noVNC/wiki/Browser-support">here</a>.

### Server Requirements

Unless you are using a VNC server with support for WebSockets
connections (such as [x11vnc/libvncserver](http://libvncserver.sourceforge.net/) or
[PocketVNC](http://www.pocketvnc.com/blog/?page_id=866)),
you need to use a WebSockets to TCP socket proxy. There is
a python proxy included ('websockify').

### Quick Start

* Use the launch script to start a mini-webserver and the WebSockets
  proxy (websockify). The `--vnc` option is used to specify the location of
  a running VNC server:

    `./utils/launch.sh --vnc localhost:5901`

* Point your browser to the cut-and-paste URL that is output by the
  launch script. Enter a password if the VNC server has one
  configured. Hit the Connect button and enjoy!

### Other Pages

* [Encrypted Connections](https://github.com/kanaka/websockify/wiki/Encrypted-Connections). How
to setup websockify so that you can use encrypted connections from noVNC.

* [Advanced Usage](https://github.com/kanaka/noVNC/wiki/Advanced-usage). Starting a VNC server,
advanced websockify usage, etc.

* [Integrating noVNC](https://github.com/kanaka/noVNC/wiki/Integration) into existing projects.

* [Troubleshooting noVNC](https://github.com/kanaka/noVNC/wiki/Troubleshooting) problems.

### Authors/Contributors

* noVNC : Joel Martin (github.com/kanaka)
    * UI and Icons : Chris Gordon
    * Original Logo : Michael Sersen
    * tight encoding : Michael Tinglof (Mercuri.ca)

* Included libraries:
    * web-socket-js : Hiroshi Ichikawa (github.com/gimite/web-socket-js)
    * as3crypto : Henri Torgemane (code.google.com/p/as3crypto)
    * base64 : Martijn Pieters (Digital Creations 2), Samuel Sieb (sieb.net)
    * jsunzip : Erik Moller (github.com/operasoftware/jsunzip),
    * tinflate : Joergen Ibsen (ibsensoftware.com)
    * DES : Dave Zimmerman (Widget Workshop), Jef Poskanzer (ACME Labs)