AVG protected virtual file server on Amazon's Elastic Cloud 2:

Starting-up and settings

Document revision 10.0 (22.6.2011)

Contents

1. Introduction

This guide is a documented step by step instruction list showing how to configure and run secured file server on EC2.

The pre-configured file server image is available on Amazon EC2 and is based on the hardened Debian GNU/Linux using Samba fileserver and AVG for Linux Free antivirus solution.

1.1. Pre-requisities

What do you need to set up the file server?

- EC2 account with access to the particular zone.

- Ability to run "SSH client" software to the image. The client is build-in on Mac and many GNU/Linux distributions. On Windows you need to download any free client, like (see links at the close of this guide), or execute a web search for "ssh client windows".

  One of the best free clients you can use is **PuTTY** (you can easily download it via link at the close of this guide). In the Useful links section at the close of this guide, there is also a short instruction for connecting to your Linux/UNIX instance from a Windows machine using this particular client.

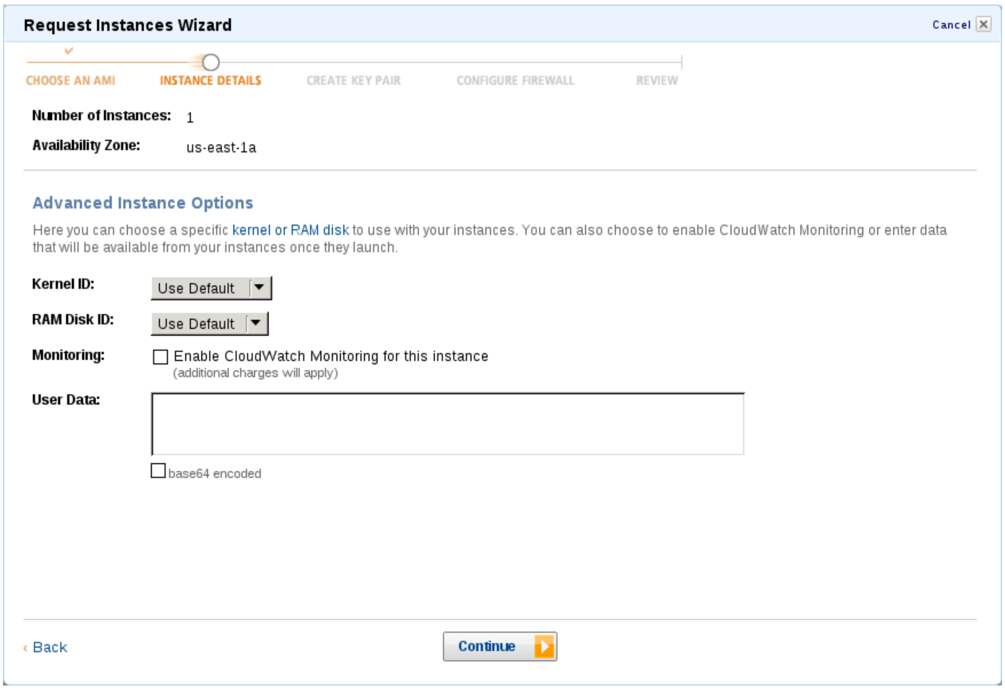
- Ability to connect to port 22 on target machine. This depends on your ISP. You probably have this ability unless you are behind some really restrictive firewall.

1.2. Running the image

AVG images are available under the name ***AVG FREE Anti-virus 8.5/ secured Debian Squeeze fileserver***. Accessibility of these images with appropriate IDs is captured by the following table:

REGION	AMI
us-east	ami-94758dfd
us-west	ami-7d4c1e38
eu-west	ami-7818290c
ap-southeast	ami-d0750d82
ap-northeast	ami-8e309a8f

You run the image like any other AMI on the EC2. The required RAM Disk and Kernel IDs are already pre-set, so just go on with default parameters.



Request Instances Wizard

Cancel ☒

CHOOSE AN AMI ✓ — INSTANCE DETAILS — CREATE KEY PAIR — CONFIGURE FIREWALL — REVIEW

Number of Instances: 1

Availability Zone: us-east-1a

Advanced Instance Options

Here you can choose a specific kernel or RAM disk to use with your instances. You can also choose to enable CloudWatch Monitoring or enter data that will be available from your instances once they launch.

Kernel ID: Use Default ▾

RAM Disk ID: Use Default ▾

Monitoring: □ Enable CloudWatch Monitoring for this instance (additional charges will apply)

User Data:

□ base64 encoded

‹ Back

Continue

Default Kernel ID is ***required***. Should you choose any other Kernel image, described functionality won't work. You should also keep in mind that Kernel IDs differ in different zones.

If you intend to attach already prepared storage volume, don't forget to run the image in the same zone, where the volume is available.

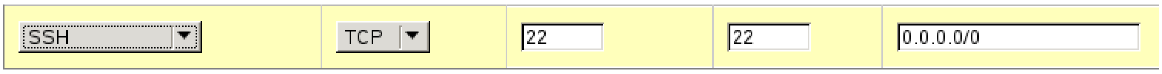
Due to security reasons, password protected login is disabled for the instance. To access the instance, you will need a key-pair. If you run the instance via AWS, you will be prompted to choose from your existing keys during the instance setup. If you don't have any key-pair, the wizard will be able to generate you one.

Once the system boots up, it starts performing initial set-up steps. Even if the AWS tells you that the system is ready for operation, you will not be able to connect for several additional minutes (which is explained in the next chapter).

1.3. Connecting to the instance

To connect to the machine, SSH protocol should be used. You have to make sure that the AWS firewall allows you to connect to the instance's port 22 (common port for the SSH protocol).

Select **Security group** in the AWS Console. You may create new security group or use your default one. If the port 22 is not yet allowed, you can simply allow it by using the bottom yellow configuration line in the security group:



SSH ▾	TCP ▾	22	22	0.0.0.0/0

Just select **SSH** in the left selector. You only need TCP communication protocol and port 22, so just leave the "TCP" and two 22's. You may also limit the IP address range of machines able to connect to your file server. While it is not necessary to adjust this setting for testing purposes, it is recommended to set it once your file server starts holding your real private data. The SSH connection is needed only for management purposes.

Once the instance is running, you need to get its IP address. You can do it by selecting the instance in AWS and clicking on **Instance actions -> Connect**.

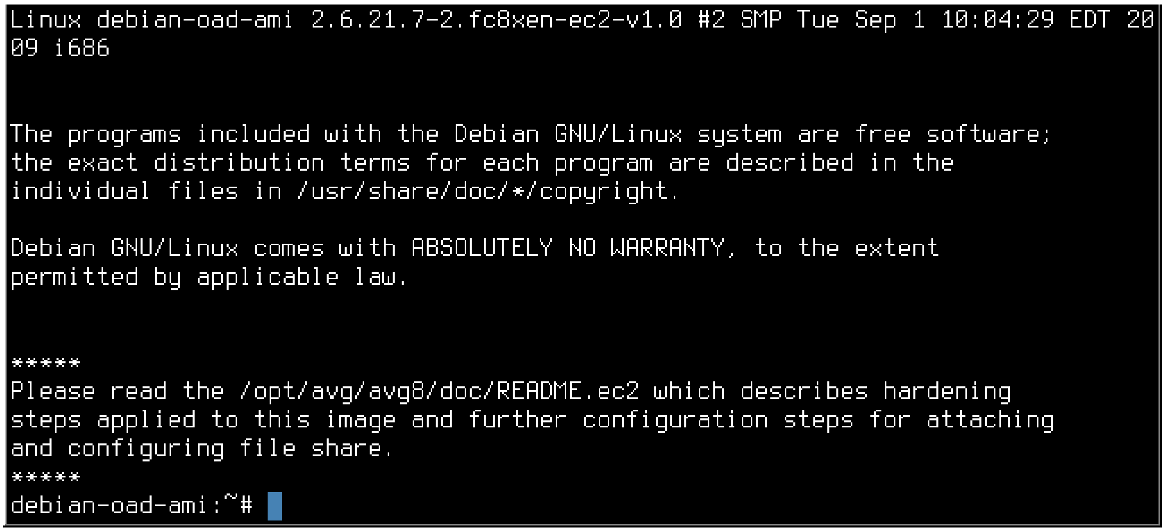
By using the IP address and the private part of the selected key-pair, you should be able to connect to the running instance. Please follow the instructions given by the AWS dialog **Instance actions -> Connect**.

See links at the close of this guide for reference on using PuTTY SSH client on Windows in order to connect to the instance. Once you are successfully connected, the instructions are similar for all possible SSH clients.

It takes some time for the instance to fully initialize. If you run the SSH client too early, you can end up with "Connection refused" error (this means that the SSH server is not yet ready) or "Access denied" error (the key is not yet in place). In such case, please retry after a few minutes.

A long period of inactivity after the SSH connection launch, which is followed by a timeout error, may indicate that you are located behind a restrictive firewall. This firewall keeps you from making a connection to the remote machine's port 22 (the port used for SSH protocol). You should contact your local administrator or ISP to allow you connection to this remote port. Running SSH client with verbose setting should help to diagnose the problem (`ssh -v -i <key>.pem root@<machine>`).

After successful connection, you should see the following:



```
Linux debian-oad-ami 2.6.21.7-2.fc8xen-ec2-v1.0 #2 SMP Tue Sep 1 10:04:29 EDT 20
09 i686


The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.


*****
Please read the /opt/avg/avg8/doc/README.ec2 which describes hardening
steps applied to this image and further configuration steps for attaching
and configuring file share.
*****
debian-oad-ami:~#
```

We recommend you to really go through the mentioned ***README.ec2*** file. Running up the system update at this point is recommended as well (see the mentioned README.

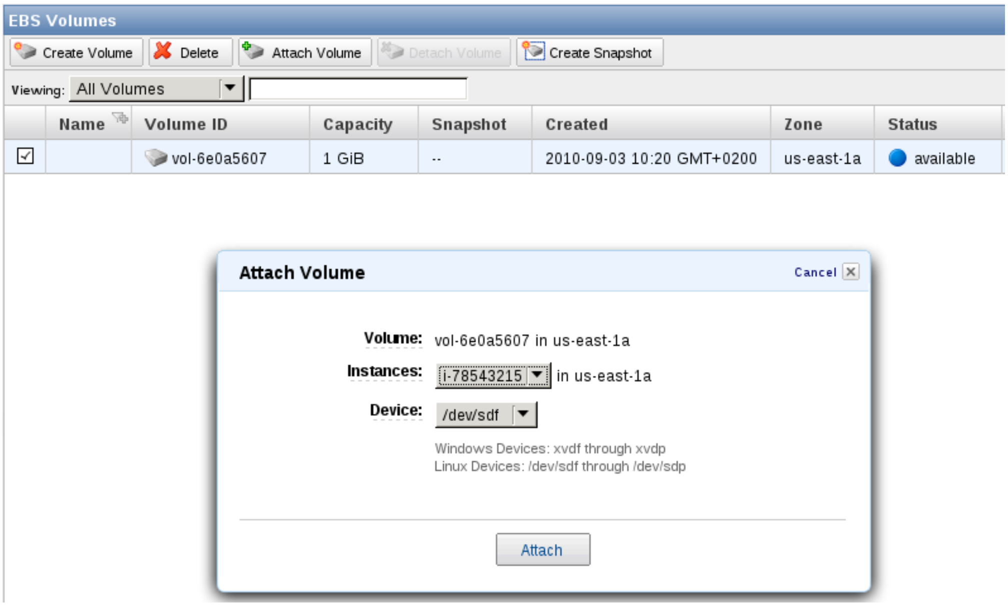
ec2 file for more details).

1.4. Attaching a volume

The following procedure is optional, but highly recommended. Of course, you may run the file server instance without a permanent storage attached (just for testing purposes), but to use it as a real file server, you will need to attach a storage.

This step consists of several sub-steps. First, you must get the storage (Elastic Block Storage) ready. If you do not have the EBS ready yet, select **Volumes** in the AWS management console and click on ***Create Volume***. Don't forget that the storage must be available in the same zone, where the instance itself is running.

Once the volume is ready, select it and click on ***Attach*** button.



	Name	Volume ID	Capacity	Snapshot	Created	Zone	Status
☑		vol-6e0a5607	1 GiB	--	2010-09-03 10:20 GMT+0200	us-east-1a	● available

Volume: vol-6e0a5607 in us-east-1a

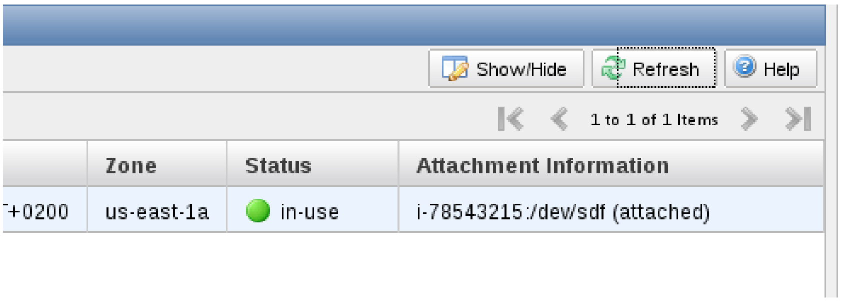
Instances: i-78543215 in us-east-1a

Device: /dev/sdf

Windows Devices: xvdf through xvdp
Linux Devices: /dev/sdf through /dev/sdp

If you have got more instances running, choose the one running file server image. You can choose any device (it is displayed in the console just in case you forget the path before you attach a device from inside the running instance).

Once you see ***attached*** in the **Attachment information** column, proceed to the next sub-step.



	Zone	Status	Attachment Information
+0200	us-east-1a	● in-use	i-78543215:/dev/sdf (attached)

Return to your instance. First, create a mount point for your storage by issuing the command:

```
mkdir /mnt/storage
```

Of course, you may use a different mount point path. If you just created the storage and this is the first time you are going to use it, you must format the file system. However, do not perform the formatting step on already formatted volume – you would loose all the data on the volume in that case!

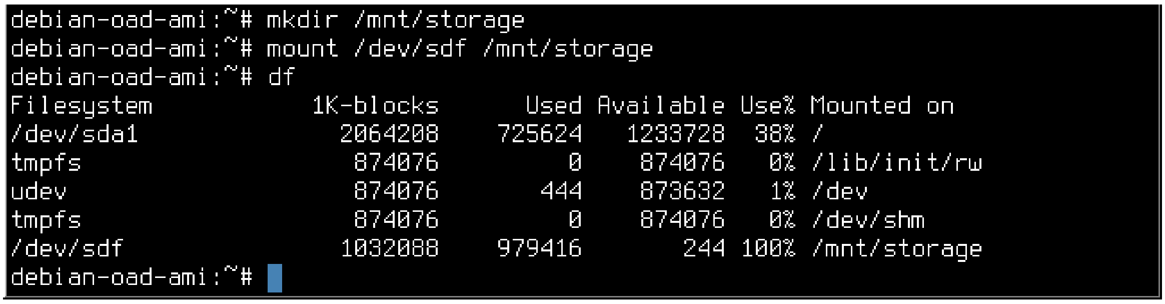
To format the volume attached to the /dev/sdf device, the following command should be issued:

```
mkfs.ext3 /dev/sdf
```

Let's attach the storage (device) to the newly created mount point now:

```
mount /dev/sdf /mnt/storage
```

Should you receive any errors, check that the device path matches the device you have chosen and that the mount point name matches the actually created mount point path. You can use the df command to check that the storage is actually attached:



```
debian-oad-ami:~# mkdir /mnt/storage
debian-oad-ami:~# mount /dev/sdf /mnt/storage
debian-oad-ami:~# df
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/sda1              2064208    725624   1233728  38% /
tmpfs                   874076         0    874076   0% /lib/init/rw
udev                    874076       444    873632   1% /dev
tmpfs                   874076         0    874076   0% /dev/shm
/dev/sdf               1032088    979416       244 100% /mnt/storage
debian-oad-ami:~#
```

You should see the /mnt/storage statistics (in our case, the device is full).

1.5. Configuring file share

To configure the file share accessible from Windows machines, we will use Samba server. There are several ways for server configuration. The preferred way is to use a text editor – the configuration is stored in the form of a text file.

Users familiar with UNIX/Linux can use the ***vi*** editor. Those unfamiliar with it should use the ***nano*** editor:

```
nano /etc/samba/smb.conf
```

The Samba server supports many options - from simple Windows sharing to a limited PDC functionality. It has several security models and is able to communicate with wide range of Windows versions, therefore it has a great deal of configuration options. Anything above setting up a basic share is outside the scope of this guide. However, you can refer to the Samba documentation (see links at the close of this guide).

You can also get a quick and quite descriptive help by looking at the manual page of samba configuration file by issuing following command:

```
man smb.conf
```

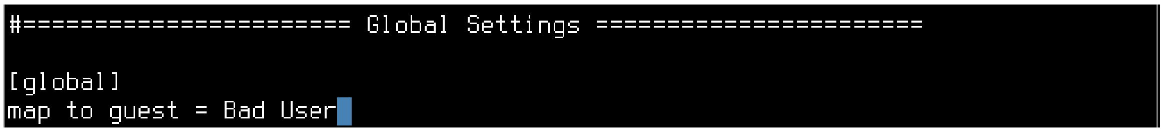
While on the manual page, use "H" key to get help on manual page navigation (searching etc.).

Let's configure a read-only public share.

When accessing any share, you need to be authenticated. To authenticate a user, Samba server must be able to identify that user against something. By default, Samba tries to match the user against a local account on the machine (so called "UNIX user"). Modern versions of Windows will try to connect using the username of actually logged-in user (which does not exist on the file server machine).

There are two options for setting up a public share. The first one is to set-up the server for "Shared security", while the second one is to preserve the user-based authentication and order Samba to map invalid users (such that do not have the appropriate "UNIX user") to the guest user.

To configure Samba for mapping invalid users to guest, you need to add the following option to the global configuration section (e.g. right after the `[global]` directive):

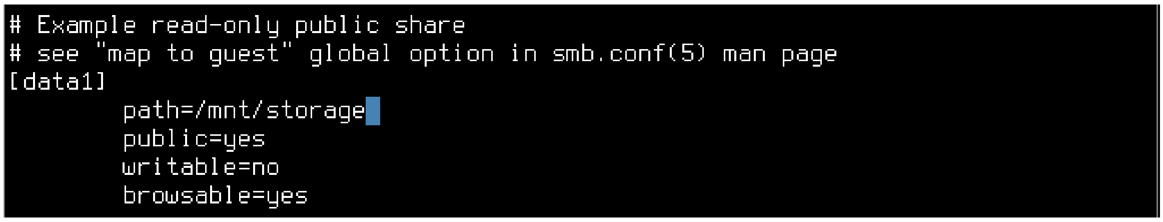


```
#======================= Global Settings =======================

[global]
map to guest = Bad User
```

Refer to the Samba man page or documentation for more information on available parameter values and their descriptions.

Now we need to configure the actual share. Scroll down to the end of the configuration file and uncomment the prepared share. Do not forget to set the share path to the actual one (`/mnt/storage` in our case):



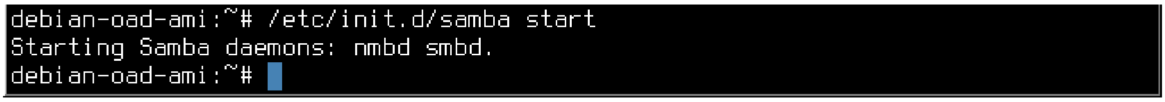
```
# Example read-only public share
# see "map to guest" global option in smb.conf(5) man page
[data1]
        path=/mnt/storage
        public=yes
        writable=no
        browsable=yes
```

Once you are finished with editing, save the file (CTRL-X followed by Y and ENTER in ***nano***).

Now you need to check your configuration with the `testparm` command from Samba suite (to check for syntax errors – just issue the `testparm` command). If you see no errors and the definition of your share is present in the `testparm` output, continue to the last configuration step – setting up the **AVG** anti-virus.

Before you start the Samba server, you may have to allow the communication ports needed for Windows file sharing (it depends if you are to run the machines inside VPC etc.). In order to do this, you need to open the **Security Group** again and allow the affected ports – 135, 137-139 and 445 depending on the version of the SMB/CIFS protocol you are using (if you are unsure, continue without these settings and return when you are unable to access the shares - more in the Testing functionality chapter).

The last sub-step is to actually start the Samba server. You may want to defer this action after the anti-virus is operational on the shared storage. Following picture displays the start command and successful output of the start action:



```
debian-oad-ami:~# /etc/init.d/samba start
Starting Samba daemons: nmbd smbd.
debian-oad-ami:~#
```

For the production server, limiting the access to the share based on hosts is highly recommended. For more info, please see Samba documentation (option "Hosts allow").

1.6. Configuring the AVG anti-virus

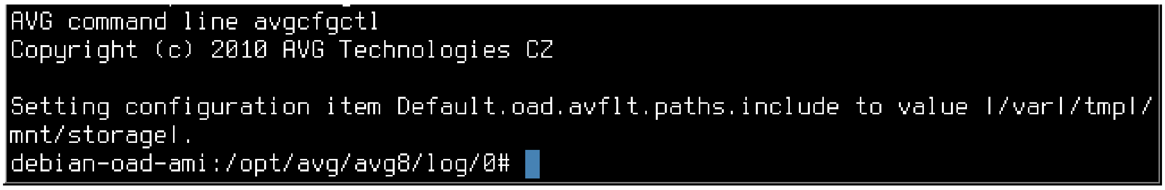
The instance is already running the on-access daemon (a part of **AVG** anti-virus for Linux Free edition). All that remains is to add the storage path to the list of paths being guarded by the on-access daemon and restart the daemon with the new configuration.

The particular configuration item holding the list of OAD guarded paths is called `Default.oad.avflt.paths.include`. **AVG** configuration is accessed using the `avgcfgctl` command.

To inspect the actual value, use `avgcfgctl  Default.oad.avflt.paths.include`. To write to the parameter, use `-w` option. See manual page (`man avgcfgctl`) for more information. Now issue the following command:

```
avgcfgctl -w  Default.oad.avflt.paths.include="|/var|/tmp|/mnt/storage|"
```

You should see this:



```
AVG command line avgcfgctl
Copyright (c) 2010 AVG Technologies CZ

Setting configuration item Default.oad.avflt.paths.include to value |/var|/tmp|/
mnt/storage|.
debian-oad-ami:/opt/avg/avg8/log/0#
```

Once configured, restart the on-access daemon:

```
avgctl --restart=oad
```

You get confirmation of success (`Operation successful.` text). You can check that the OAD was successfully restarted by inspecting the log file (`/opt/avg/avg8/log/0/oad.pub.rollog`) as well.

1.7. Checking functionality

In this chapter, you will learn to check if the **AVG** is up-to-date and online, as well as if the Samba server is running and defined share is accessible.

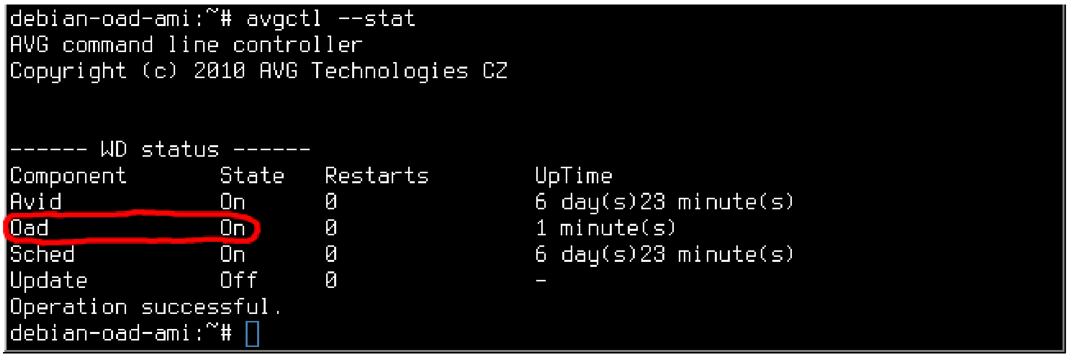
1.7.1. Checking the AVG anti-virus

How to check that the **AVG** is up-to-date and online?

- ***Run*** `avgupdate` from the command line

  AVG anti-virus should immediately perform update. Textual confirmation should be displayed.

- ***Check*** that the OAD is running by executing `avgctl --stat` and examining its output:



```
debian-oad-ami:~# avgctl --stat
AVG command line controller
Copyright (c) 2010 AVG Technologies CZ


------ WD status ------
Component       State   Restarts        UpTime
Avid            On      0               6 day(s)23 minute(s)
Oad             On      0               1 minute(s)
Sched           On      0               6 day(s)23 minute(s)
Update          Off     0               -
Operation successful.
debian-oad-ami:~#
```

  If OAD is not running, start it using `avgctl --start=oad` and retry the `avgctl --stat` after a while. If you still see off state, inspect log file (`/opt/avg/avg8/log/0/oad.pub.rollog`) for potential problems.

1.7.2. Checking the Samba server

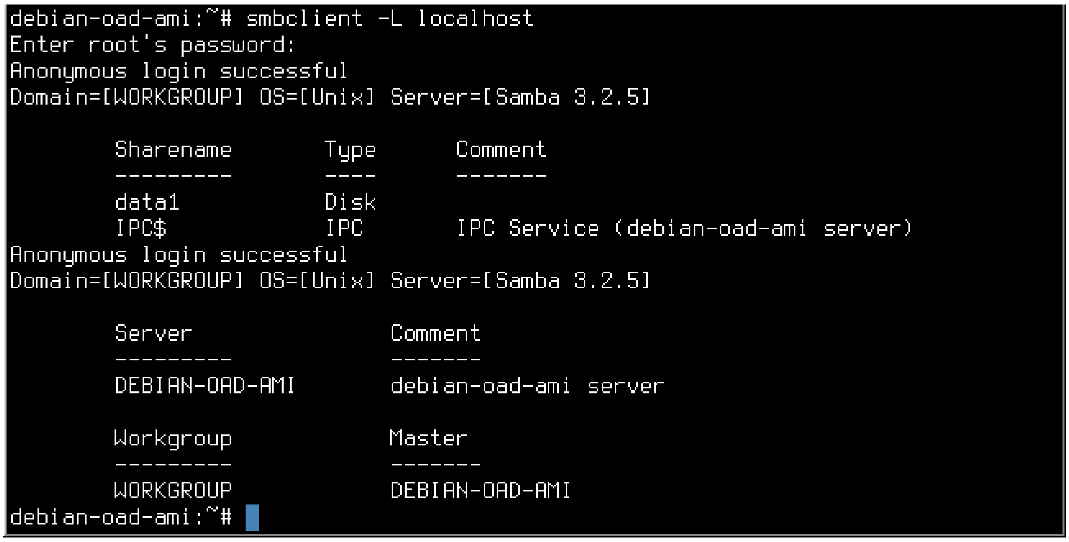
How to check that the Samba server is running and defined share is accessible?

- ***Run*** `ps aux | grep [s]mbd`

  You should see one or more statistics on process named `smbd`. Empty output means the Samba server is not running. Start it using `/etc/init.d/samba start` and retry. Consider inspecting log file (`/var/log/samba/log.smbd`) for potential problems.

- ***Use*** `smbclient` for basic check of share accessibility

  Issuing `smbclient -L localhost` should produce the following:



```
debian-oad-ami:~# smbclient -L localhost
Enter root's password:
Anonymous login successful
Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.2.5]

        Sharename       Type      Comment
        ---------       ----      -------
        data1           Disk
        IPC$            IPC       IPC Service (debian-oad-ami server)
Anonymous login successful
Domain=[WORKGROUP] OS=[Unix] Server=[Samba 3.2.5]

        Server               Comment
        ---------            -------
        DEBIAN-OAD-AMI       debian-oad-ami server

        Workgroup            Master
        ---------            -------
        WORKGROUP            DEBIAN-OAD-AMI
debian-oad-ami:~#
```

- Use empty password for authentication. The above shows that the `data1` share is publicly visible.

- Use following command to connect to the `data1` share:

  ```
  smbclient //localhost/data1
  ```

- Use `cd` and `ls` commands to inspect the content of the share (it is actually the attached storage so you should see the same content as accessible under `/mnt/storage`).

1.8. Checking share accessibility from Windows

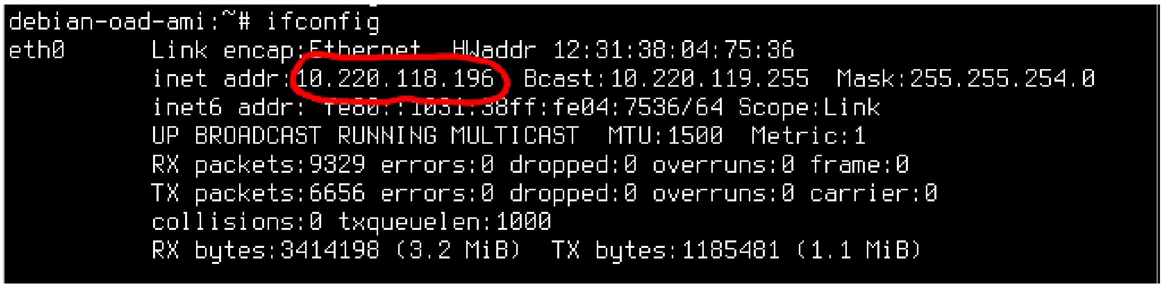
To perform share accessibility check from a Windows machine, you need to run instance of any Windows-based AMI. When running this AMI, don't forget to add the RDP port to the security group to be able to access such instance.

After you run Windows instance, select it in the AWS console and click **Instance Actions -> Get Windows Admin Password**.

Once you obtain the password, use `rdesktop` command to connect to the Windows instance (from Linux machine):

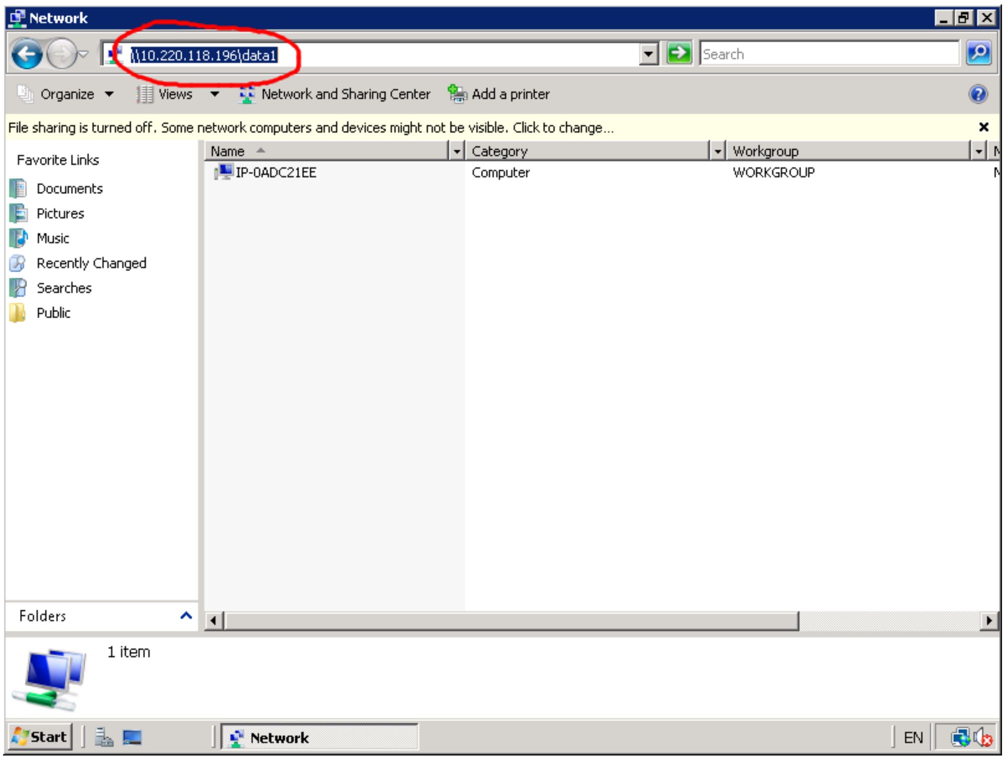
```
rdesktop -K -u Administrator -p "<password>" <Windows-machine>
```

Issue the `ifconfig` command in the file server command line to get the IP address. Look for the address following the `eth0` interface:



```
debian-oad-ami:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 12:31:38:04:75:36
          inet addr:10.220.118.196  Bcast:10.220.119.255  Mask:255.255.254.0
          inet6 addr: fe80::1031:38ff:fe04:7536/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:9329 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6656 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3414198 (3.2 MiB)  TX bytes:1185481 (1.1 MiB)
```

When you connect to the Windows instance, click on **Start -> Network** and type the IP address and share name into the top address box:



\\10.220.118.196\data1

File sharing is turned off. Some network computers and devices might not be visible. Click to change...

Name	Category	Workgroup
IP-0ADC21EE	Computer	WORKGROUP

1 item

After pressing ENTER, you should see the content of the storage. If not, inspect the Samba log files under `/var/log/samba` directory on the file server. You can use the `ls -ltr` command to see which log files were modified recently. You can set the global ***log level*** parameter to higher number to see more information in the log files (refer to manual page of `smb.conf`). If you do not see any connection in the Samba log files, test accessibility of the file server from the Windows machine. Try using `ping` first (**Start -> Command prompt**, enter `ping <address>`). You may need to explicitly allow UDP and TCP ports 135, 137-139 and 445 on the file server (in the **Security Group**), depending on the network setup.

1.9. Checking anti-virus functionality using the test file

The **AVG** anti-virus on-access daemon is running in the background and intercepting any access to the file (every file open). Once it identifies an infected file, it blocks access to that file – even for the root user.

The process accessing the infected file gets Permission Denied error, irrespective of the actual permission set.

You can easily test that the anti-virus is blocking infected files from being accessed. To achieve this, simply put an anti-virus test file on the storage.

1. ***Download the anti-virus test file (Eicar)*** - you can use the link at the close of this guide.

2. ***Put this file on the shared storage.***

   You can also use following commands to install a `wget` software on the machine and download the eicar.com in one step:

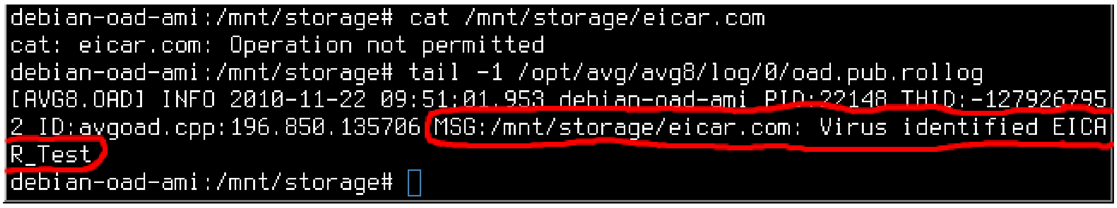
   I. `apt-get install wget`

   II. `cd /mnt/storage`

   III. `wget https://secure.eicar.org/eicar.com`

3. ***Make sure that the AVG OAD is started and operational*** (issue `avgctl --stat` command and check that OAD status says on)

4. ***Try accessing the file locally*** – use a `cd` command to move to the directory containing the file, then issue the `cat eicar.com` command (substitute the file name if your sample has a different name). Optionally, you can check the log file of OAD, saying that the Eicar test sample was identified:



```
debian-oad-ami:/mnt/storage# cat /mnt/storage/eicar.com
cat: eicar.com: Operation not permitted
debian-oad-ami:/mnt/storage# tail -1 /opt/avg/avg8/log/0/oad.pub.rollog
[AVG8.OAD] INFO 2010-11-22 09:51:01.953 debian-oad-ami PID:22148 THID:-127926795
2 ID:avgoad.cpp:196.850.135706 MSG:/mnt/storage/eicar.com: Virus identified EICA
R_Test
debian-oad-ami:/mnt/storage#
```

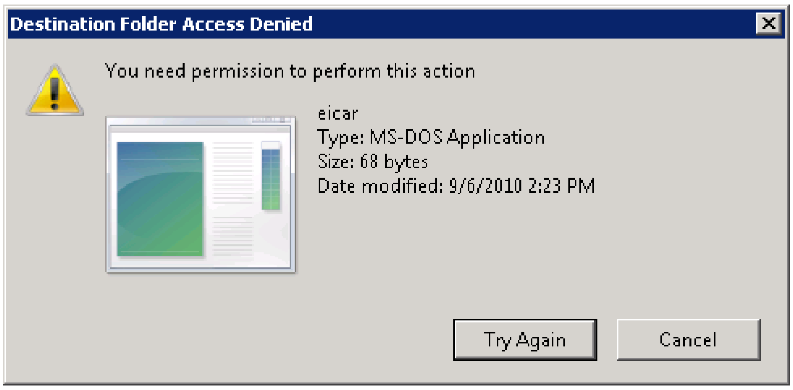
   Note that (as it is shown in the screenshot above) the command used for displaying the last line of the on-access daemon log file is:

   ```
   tail -1 /opt/avg/avg8/log/0/oad.pub.rollog
   ```

5. ***Finally, try to access the share.***

   First, connect to the share.

Now, simple clicking on the infected file (Eicar) from Windows will probably produce no effect (depending on system version, Windows may not show the actual error message), but copying and pasting the file to another location should bring up the error dialog:



Destination Folder Access Denied

You need permission to perform this action

eicar
Type: MS-DOS Application
Size: 68 bytes
Date modified: 9/6/2010 2:23 PM

While the error says **Destination Folder**, it actually refers to the permission error given by Samba server (which in turn receives this error from system when trying to access the file).

1.10. Further actions

The following actions are optional, but quite purposeful, so you may consider to perform them:

- ***Installing more software on the file server machine***

  The instance is configured to work with standard Debian repository. You should prefer installing the software directly from the Debian repository when possible, due to easy way of managing and the fact that all the software in the repository was tested with the operation system the instance is running.

  Use `aptitude` or `apt-get` commands. Refer to Debian documentation (see links at the close of this guide).

  Generally, be cautious when installing software. Additional software may bring security issues.

  Installing additional software (but also adding new users) may require enabling the ***root account***. This account is disabled by default; the only way to log in as root is to use key authentication via SSH. This is why it is highly recommended to set the root password (which remains empty by default).

- ***Setting up a firewall***

  Please bear in mind that access to the machine is protected via **Security Group** setting on Amazon side. This means that unless you have very specific demands, there is no need for you to set a firewall up using the IP tables.

- ***Further hardening steps***

  Read the README.ec2 file bundled with the file server image:

  ```
  /opt/avg/avg8/doc/README.ec2
  ```

- ***Further anti-virus configuration***

  Read the AVG for Linux documentation available at:

  o files under the `/opt/avg/avg8/doc/` directory

  o manual pages

  o user documentation in PDF (see links at the close of this guide)

2. Useful links

- ***Putty download:***

  http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html

- ***Putty on EC2:***

  http://docs.amazonwebservices.com/AWSEC2/latest/GettingStartedGuide/index.html?ConnectToInstanceLinux.html#d0e515

- ***Samba documentation:***

  http://www.samba.org/samba/docs/

- ***Debian GNU/Linux documentation:***

  http://www.debian.org/doc/

- ***AVG for Linux documentation:***

  http://download.avg.com/filedir/doc/LINUX_GROUP/AVG_Anti-Virus_for_Linux/avg_avl_uma_en_85_2.pdf

- ***Eicar donwload page:***

  http://www.eicar.org/anti_virus_test_file.htm

- ***Amazon webpages:***

  - FAQ section: http://aws.amazon.com/ec2/faqs/
  - Getting started with Amazon's Elastic Cloud 2: http://docs.amazonwebservices.com/AWSEC2/latest/GettingStartedGuide/
  - User guide for Amazon's Elastic Cloud 2: http://docs.amazonwebservices.com/AWSEC2/latest/UserGuide/