1. Follow the instructions by Richard Miller to 9fans to create a new instance: http://9fans.net/archive/2012/05/453

   > *From: Richard Miller <9fans@ham...>*
   > *Subject: plan 9 in the cloud - amazon ec2*
   > *Date: Thu, 24 May 2012 08:47:42 +0100*
   >
   > *I've made a few tweaks to the paravirtualised plan 9 kernel in /n/sources/xen/xen3 to make it compatible with the xen environment used in amazon's "elastic compute cloud". There's an example public AMI (in zone eu-west-1) named ami-03c9f377 which can be used to instantiate a cpu server with fossil on a 1GB "elastic block storage" volume. This fits within the free usage tier which you get for the first 12 months after creating an amazon web services account.*
   >
   > *When first launched, your server will start with authid = ec2, authdom and sysname equal to the ec2 instance id, and a randomly generated password which will be echoed to the system console, where it can be retrieved - only by the instance's owner - using the Get System Log function of the aws web interface. (Be patient; there is often a delay of some minutes after launching an instance before the console log becomes available.)*
   >
   > *Alternatively, if the "user data" field, which you can create as part of launching an ec2 instance, contains a shell script (anything following a line beginning with #!/bin/rc), this will be run from /bin/cpurc.local at boot time. You can use this to inject your own initial authentication details into factotum, by putting something like this in the user data field:*
   >
   > ```
   >   #!/bin/rc
   >   auth/factotum -g 'proto=p9sk1 user=ec2 dom=my.auth.dom !password=XXXX'
   > ```
   >
   > *The random password or factotum script will allow you to connect to your server for the first time using cpu, drawterm or ssh1. (For the first two, don't forget to open port 17010 or 17013 in the "security group" firewall.) You'll then want to use auth/wrkey to put new credentials securely into nvram, and reboot or run 'auth/readnvram >/mnt/factotum/ctl' so they can take effect. Afterwards you can delete the old temporary password from user data, but only when the instance is stopped.*
   >
   > *For the curious, I've set up a plan 9 instance at ec2.hamnavoe.com, using authentication domain outside.plan9.bell-labs.com so anyone with a sources account can cpu into it. I'll leave it running for a few days (until my monthly free usage tier quota runs out).*

2. To create a new kernel, follow the instructions in /n/sources/xen/xen3/README.
3. Create a new kernel (e.g 9xenpccpufrb.gz).
4. The kernel doesn't go in /386 but in a separate ext3 partition where it can be booted by pvgrub. The ext3 is not mounted by default, and might have to run /bin/diskparts before, in order to see it (check /dev/sd00 first to see if the 'linux' partition exists, that's the partition we want).

5. Mount the ext3 partition on /n/linux and copy the kernel to /n/linux/boot. usually ext2srv is running so it might be as easy as:

   ```
   mount /srv/ext2 /n/linux /dev/sd00/linux
   ```

6. Change the default option in /n/linux/boot/grub/default to 2, and change /n/linux/boot/grub/menu.lst to something like:

   ```
   # edit grub/default file to select configuration for next boot
   default saved
   timeout 0
   hiddenmenu

   title Plan 9 fossil
   root  (hd0,0)
   kernel  /boot/9xenpccpuf.gz nobootprompt=local!#S/sd00/fossil
   venti=127.0.0.1

   title Plan 9 tcp
   root  (hd0,0)
   kernel  /boot/9xenpccpuf.gz nobootprompt=tcp fs=N.N.N.N auth=N.N.N.N

   # test configuration
   #   'savedefault' reverts to fallback configuration on subsequent
   reboot title Plan 9 test
   root  (hd0,0)
   kernel  /boot/9xenpccpufrb.gz nobootprompt=local!#S/sd00/fossil
   venti=127.0.0.1
   savedefault 0
   ```

7. Once you’ve verified that the new one boots ok, make it the default (i.e. 0) kernel.