Passwordless root access in qubes — Qubes Docs

6-8 minutes

Background (`/etc/sudoers.d/qubes` in VM):

```
user ALL=(ALL) NOPASSWD: ALL

# WTF?! Have you lost your mind?!
#
# In Qubes VMs there is no point in isolating the root account from
# the user account. This is because all the user data is already
# accessible from the user account, so there is no direct benefit for
# the attacker if she could escalate to root (there is even no benefit
# in trying to install some persistent rootkits, as the VM's root
# filesystem modifications are lost upon each start of a VM).
#
# One might argue that some hypothetical attacks against the
# hypervisor or the few daemons/backends in Dom0 (so VM escape
# attacks) most likely would require root access in the VM to trigger
# the attack.
#
# That's true, but mere existence of such a bug in the hypervisor or
# Dom0 that could be exploited by a malicious VM, no matter whether
# requiring user, root, or even kernel access in the VM, would be
# FATAL. In such situation (if there was such a bug in Xen) there
# really is no comforting that: "oh, but the mitigating factor was
# that the attacker needed root in VM!" We're not M$, and we're not
# gonna BS our users that there are mitigating factors in that case,
# and for sure, root/user isolation is not a mitigating factor.
#
# Because, really, if somebody could find and exploit a bug in the Xen
# hypervisor -- as of 2016, there have been only three publicly disclosed
# exploitable bugs in the Xen hypervisor from a VM -- then it would be
# highly unlikely if that person couldn't also found a user-to-root
# escalation in VM (which as we know from history of UNIX/Linux
# happens all the time).
#
# At the same time allowing for easy user-to-root escalation in a VM
# is simply convenient for users, especially for update installation.
#
# Currently this still doesn't work as expected, because some idotic
# piece of software called PolKit uses own set of policies. We're
# planning to address this in Beta 2. (Why PolKit is an idiocy? Do a
# simple experiment: start 'xinput test' in one xterm, running as
```

```
# user, then open some app that uses PolKit and asks for root
# password, e.g.  gpk-update-viewer -- observe how all the keystrokes
# with root password you enter into the "secure" PolKit dialog box can
# be seen by the xinput program...)
#
# joanna.
```

Below is a complete list of configuration made according to the above statement, with (not necessary complete) list of mechanisms depending on each of them:

1. sudo (`/etc/sudoers.d/qubes`):

   ```
   user ALL=(ALL) NOPASSWD: ALL
   (...)
   ```

   - Easy user -> root access (main option for the user).
   - `qvm-usb` (not really working, as of R2).

2. PolicyKit (`/etc/polkit-1/rules.d/00-qubes-allow-all.rules`):

   ```
   //allow any action, detailed reasoning in sudoers.d/qubes
   polkit.addRule(function(action,subject) { return polkit.Result.YES;
   });
   ```

   and `/etc/polkit-1/localauthority/50-local.d/qubes-allow-all.pkla`:

   ```
   [Qubes allow all]
   Identity=*
   Action=*
   ResultAny=yes
   ResultInactive=yes
   ResultActive=yes
   ```

   - NetworkManager configuration from normal user (`nm-applet`).
   - Updates installation (`gpk-update-viewer`).
   - User can use pkexec just like sudo Note: above is needed mostly because Qubes user GUI session isn't treated by PolicyKit/logind as "local" session because of the way in which X server and session is started. Perhaps we will address this issue in the future, but this is really low priority. Patches welcomed anyway.

3. Empty root password:

   - Used for access to 'root' account from text console (`qvm-console-dispvm`) - the only way to access the VM when GUI isn't working.
   - Can be used for easy 'su -' from user to root.

Replacing passwordless root access with Dom0 user prompt¶

While ITL supports the statement above, some Qubes users may wish to enable user/root isolation in VMs anyway. We do not support it in any of our packages, but of course nothing is preventing the user from modifying his or her own system. A list of steps to do so is provided here **without any guarantee of safety, accuracy, or completeness. Proceed at your own risk. Do not rely on this for extra security.**

1. Adding Dom0 “VMAuth” service:

   ```
   [root@dom0 /]# echo "/usr/bin/echo 1" >/etc/qubes-rpc/qubes.VMAuth
   [root@dom0 /]# echo "@anyvm dom0 ask,default_target=dom0" \
   >/etc/qubes-rpc/policy/qubes.VMAuth
   [root@dom0 /]# chmod +x /etc/qubes-rpc/qubes.VMAuth
   ```

   (Note: any VMs you would like still to have passwordless root access (e.g. Templates) can be specified in the second file with “<vmname> dom0 allow”)

2. Configuring Fedora template to prompt Dom0 for any authorization request:

   - In `/etc/pam.d/system-auth`, replace all lines beginning with “auth” with these lines:

     ```
     auth  [success=1 default=ignore]  pam_exec.so seteuid /usr/lib/
     qubes/qrexec-client-vm dom0 qubes.VMAuth /bin/grep -q ^1$
     auth  requisite  pam_deny.so
     auth  required   pam_permit.so
     ```

   - Require authentication for sudo. Replace the first line of `/etc/sudoers.d/qubes` with:

   - Disable PolKit’s default-allow behavior:

     ```
     [root@fedora-20-x64]# rm /etc/polkit-1/rules.d/00-qubes-allow-
     all.rules
     [root@fedora-20-x64]# rm /etc/polkit-1/localauthority/50-local.d/
     qubes-allow-all.pkla
     ```

3. Configuring Debian/Whonix template to prompt Dom0 for any authorization request:

   - In `/etc/pam.d/common-auth`, replace all lines beginning with “auth” with these lines:

     ```
     auth  [success=1 default=ignore]  pam_exec.so seteuid /usr/lib/
     qubes/qrexec-client-vm dom0 qubes.VMAuth /bin/grep -q ^1$
     auth  requisite  pam_deny.so
     auth  required   pam_permit.so
     ```

   - Require authentication for sudo. Replace the first line of `/etc/sudoers.d/qubes` with:

- Disable PolKit's default-allow behavior:

```
[root@debian-8]# rm /etc/polkit-1/rules.d/00-qubes-allow-all.rules
[root@debian-8]# rm /etc/polkit-1/localauthority/50-local.d/qubes-allow-all.pkla
```

- In `/etc/pam.d/su.qubes`, comment out this line near the bottom of the file:

```
auth sufficient pam_permit.so
```

- For Whonix, if prompts appear during boot, create `/etc/sudoers.d/zz99` and add these lines:

```
ALL ALL=NOPASSWD: /usr/sbin/virt-what
ALL ALL=NOPASSWD: /usr/sbin/service whonixcheck restart
ALL ALL=NOPASSWD: /usr/sbin/service whonixcheck start
ALL ALL=NOPASSWD: /usr/sbin/service whonixcheck stop
ALL ALL=NOPASSWD: /usr/sbin/service whonixcheck status
```

Dom0 passwordless root access¶

There is also passwordless user->root access in dom0. As stated in comment in sudo configuration there (different one than VMs one), there is really no point in user/root isolation, because all the user data (and VM management interface) is already accessible from dom0 user level, so there is nothing more to get from dom0 root account.