Ovirt 3.6 deep dive: guest serial console

Francesco Romani
Software Engineer
Red Hat Inc.

Serial Console (1/2)

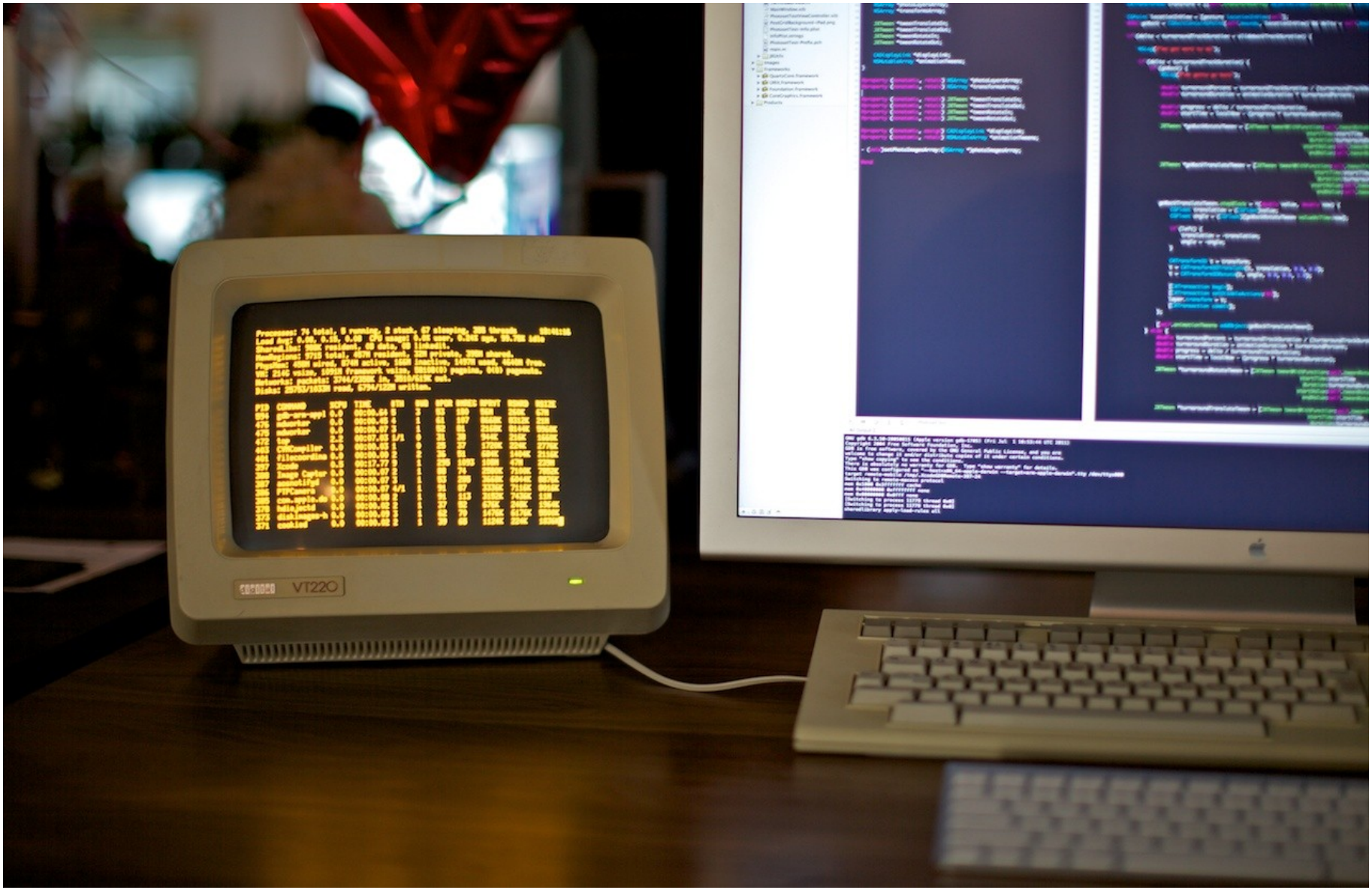


Image credits: http://40.media.tumblr.com/tumblr_lpo5dfqyIz1qz4mo8o1_r1_1280.jpg

Serial Console (2/2)

Why use a serial console?

Systems administration of remote computers

> [...]
> System administration of these remote computers is usually done using SSH, but there are times when access to the console is the only way to diagnose and correct software failures. Major upgrades to the installed distribution may also require console access.
> [...]

Quoted from: http://www.tldp.org/HOWTO/Remote-Serial-Console-HOWTO/intro-why.html

VM Serial Console (1/4)

root@shinji:/root

File Edit View Search Terminal Help

```
SHINji> 12:10:54 root [~]$ virsh console 3
Connected to domain serialconsole2-c7x
Escape character is ^]

CentOS Linux 7 (Core)
Kernel 3.10.0-229.7.2.el7.x86_64 on an x86_64

c7 login: root
Password:
Last login: Wed Jul 22 10:39:47 on pts/1
[root@c7 ~]# w
 06:11:08 up  1:23,  1 user,  load average: 0.00, 0.01, 0.03
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
root     ttyS0                     06:11    4.00s  0.02s  0.02s w
[root@c7 ~]#
```

VM Serial Console (2/4)

Activities Firefox

oVirt Engine Web A

oVirt Engine Web Ad...

192.168.1.198:8080/ovirt-engine/webadmin/?locale=en_US#vms-general

oVirt OPEN VIRTUALIZATION MANAGER

Vms:

Data Centers | Clusters | Hosts | Networks | Storage | Disks | Virtual Machines | Pools

System

Expand All Collapse All

System

New VM Import Edit Remove Clone VM Run Once Migrate Cancel Migration Make Template Export

Name	Comment	Host	IP Address	FQDN	Cluster
sercon_test-1		kenji			C7

Edit Virtual Machine

- High Availability
- Resource Allocation
- Boot Options
- Random Generator
- Custom Properties
- Icon

Optimized for: Server

Video Type: QXL

Graphics protocol: SPICE

USB Support: Disabled

Console Disconnect Action: Lock screen

Monitors: 1 — Single PCI

Smartcard Enabled

Single Sign On method

Disable Single Sign On

Use Guest Agent

Advanced Parameters

Soundcard enabled

>>>>>> VirtIO Console Device Enabled

Enable SPICE file transfer

Enable SPICE clipboard copy and paste

Hide Advanced Options | OK | Cancel

Vm Devices

Defined Memory:

Physical Memory

Guest OS Memor

Number of CPU C

Guest CPU Count

VM Serial Console (3/4)

Accessing the VM Serial console in oVirt <= 3.5

1. Find the host on which the VM is running
    1. connect to ovirt-engine webadmin
    2. lookup in the VM page
2. Connect using SSH to the host
    1. ssh -i ident.key hypervisor-host
3. Find the libvirt ID of the VM
    1. virsh list | less
4. Use virsh to connect to the VM console
    1. Virsh console $VM_ID
    2. You must use the VDSM auth

VM Serial Console (4/4)

Accessing the VM Serial console in oVirt 3.6

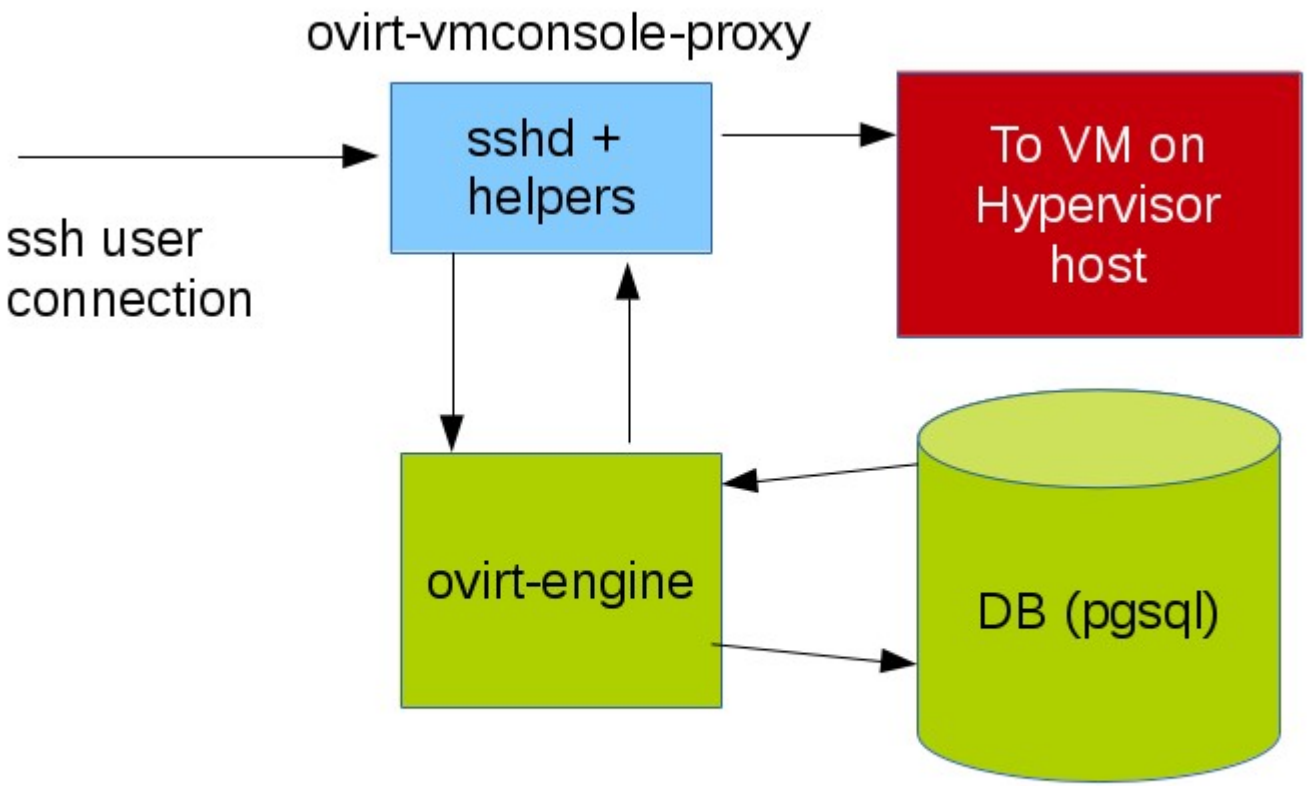
1. **ssh -i console.key -p 2222 -t ovirt-vmconsole@proxy-host**

That's it!

Key design points (1/3)

Proxy server

- No direct connection to Vms



Key design points (2/3)

- External generic ovirt-vmconsole package
  - ovirt-vmconsole-proxy runs on the user-facing host
  - ovirt-vmconsole-host runs on each virtualization host

Key design points (3/3)

- ovirt-engine stores user and VM data (positioning, identifiers)
- ovirt-engine can optionally integrate with ovirt-vmconsole
  - Integration is enabled by default on ovirt-engine-setup if the ovirt-vmconsole package is detected

Ovirt-vmconsole-proxy from 10k ft

- ovirt-vmconsole-proxy configures a special-purpose sshd instance
- sshd does all the transport-related duties
- external tools must provide
  - SSH keys storage and retrieval
  - VM positioning information
  - ovirt-engine can obviously and easily provide both.

Role of Ovirt Engine (1/2)

- ovirt-vmconsole-proxy asks ovirt-engine for all the authentication keys
- To use the vmconsole proxy, one user must register the SSH key in engine
- The user must be able to login in Engine
  - Integration point: ovirt-vmconsole-proxy-keys

Role of Ovirt Engine (1/2)

- ovirt-engine knows if and where a given VM is running
- ovirt-engine knows if an user has permission to connect to a VM
- ovirt-vmconsole-proxy asks ovirt-engine for available Vms for a given user
  - Integration point: ovirt-vmconsole-proxy-shell

Engine Key Management

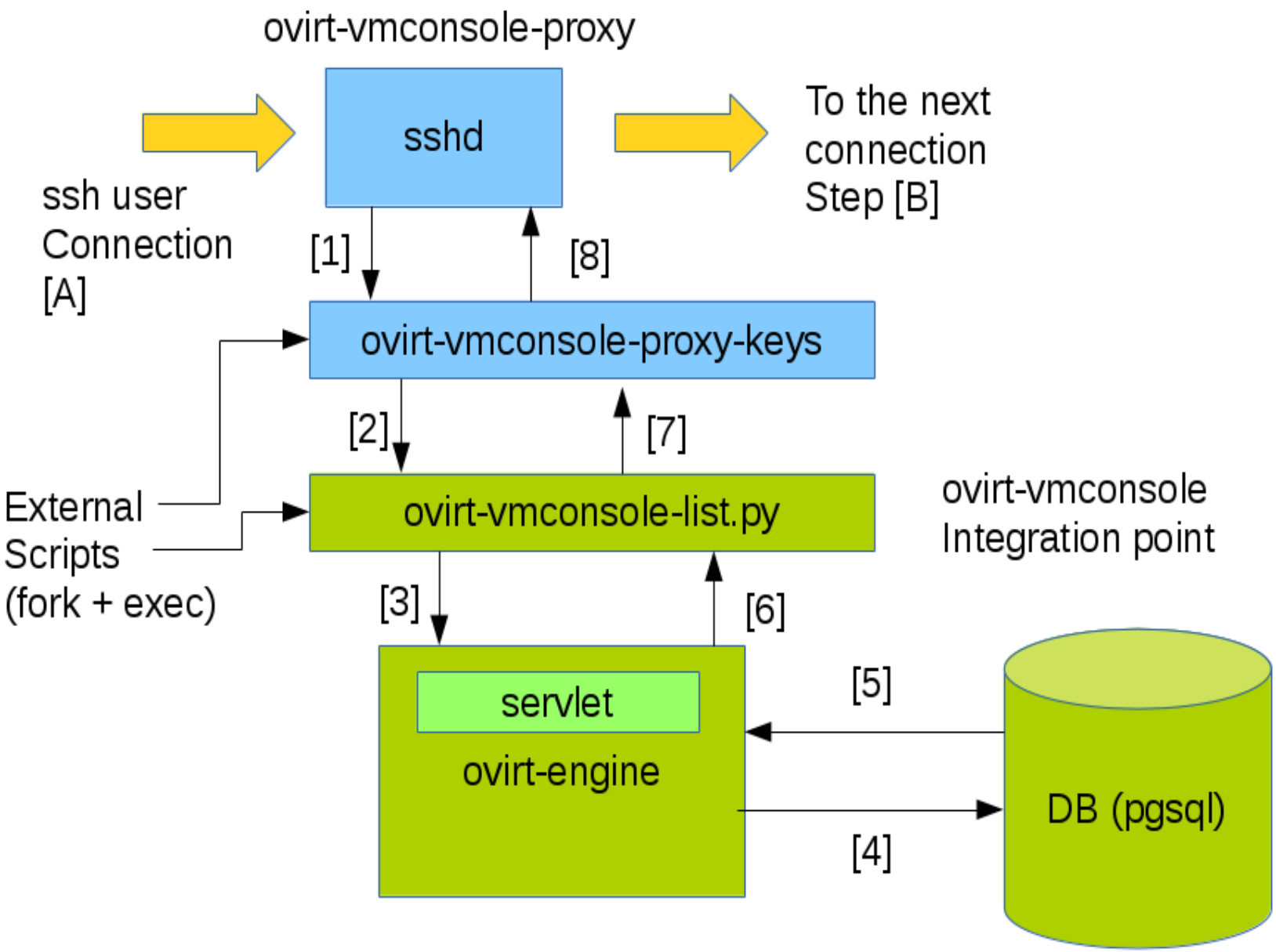
- ovirt-engine stores the SSH public keys of the users
  - Currently only one key per user [*]
  - Replaces $HOME/.ssh/authorized_keys

Ovirt-vmconsole-proxy flow (1/2)

1. On connection attempt:
2. Special-configured sshd asks ovirt-engine for the list of all known keys
    1. Uses ovirt-vmconsole-proxy-keys helper
    2. The ovirt-vmconsole package is generic: use ovirt-engine specific script (shipped with engine)
3. On succesfull authentication, runs another helper:

   ovirt-vmconsole-proxy-shell
4. Fetches a list of available VM consoles from Engine
    1. Present the list to the user, allows to select a VM to connect to
    2. If the user specified a VM, validates it against the list then connect

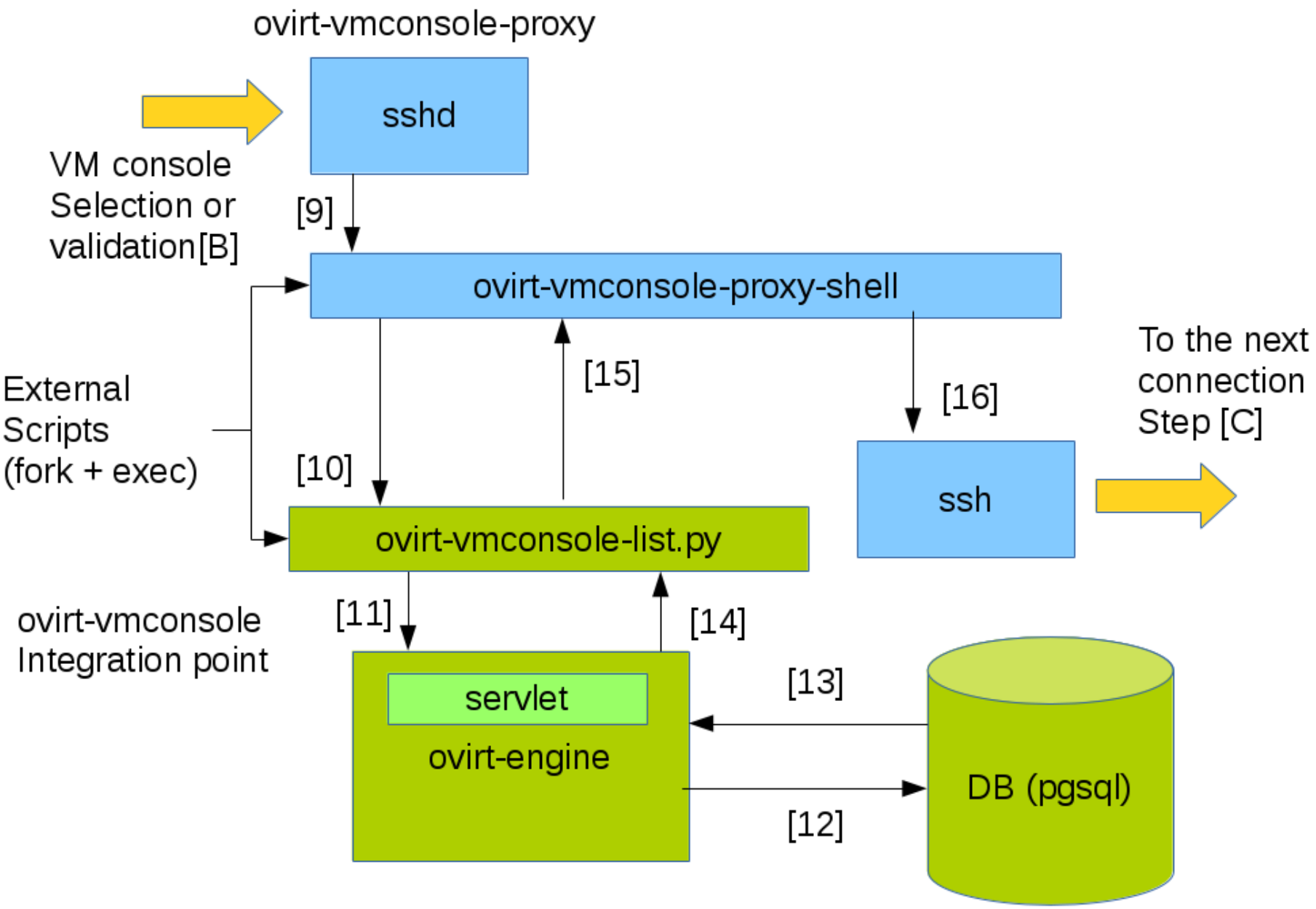
Ovirt-vmconsole-proxy flow (2/2)



Ovirt-vmconsole-host (1/2)

1. Once the user succesfully authenticated on the proxy and succesfully selected a VM, we must jump on it
2. The ovirt-vmconsole-host package uses special-purpose sshd instance on each hypervisor host
3. Additional SSH link between the proxy host and the hypervisor host
    1. Transparently instaurated by ovirt-vmconsole-proxy-shell
    2. Key management completely handled by ovirt-vmconsole
    3. Key enrollment handled by ovirt-host-deploy
4. Completely transparent to the user
    1. The user explicitely connect only on the proxy host

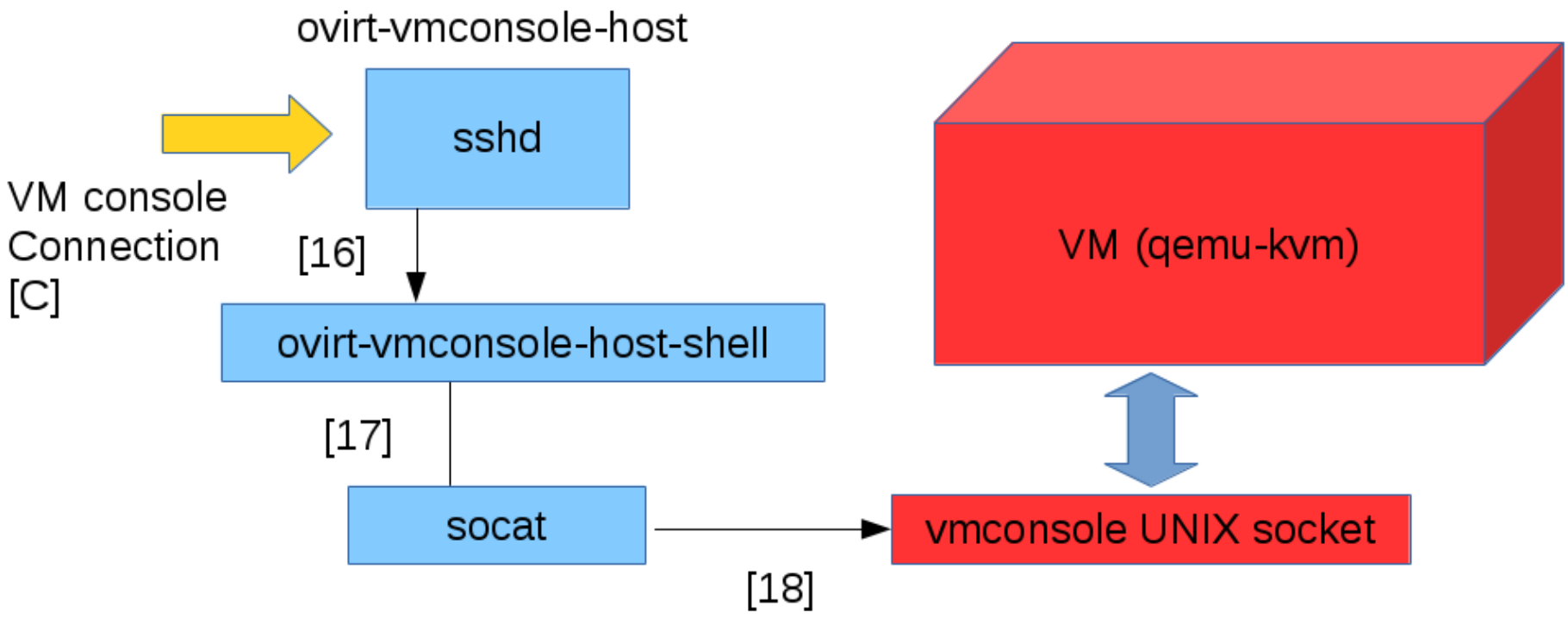
Ovirt-vmconsole-host (2/2)



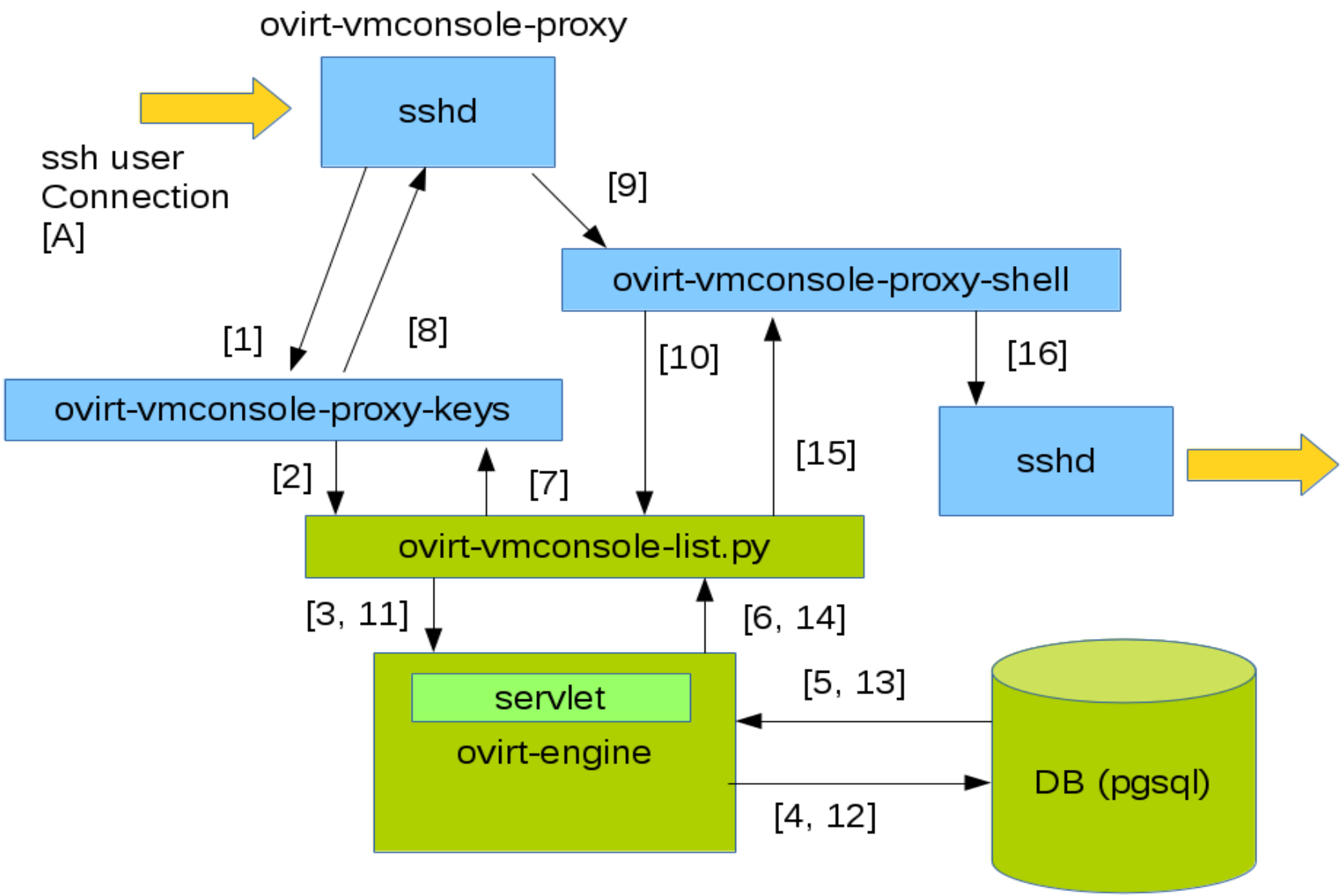
VDSM bridge (1/2)

- “Last mile” on the hypervisor host
- Default setting: bind the VM serial console to one PTY
  - This is what virsh expects and uses
  - VDSM adds very basic password protection
- VDSM binds the VM serial console to an UNIX domain socket
  - UNIX permissions and SELinux contexts guarantee security greater or equal to the password “protection” previously used
  - Virsh no longer available to connect to the VM console
  - Manual connection still possible! But one must use socat or minicom on the UNIX domain socket

VDSM bridge (2/2)



Flow summary



Current limitations

- No seamless migration support
- Proxy host must run on the same host of Engine
  - Limitations mostly on the automated setup
- Proxy depends on Engine
  - Not real limitation
  - But still, no new connections if Engine is down

Future extensions

- Allow to run proxy and Engine on different hosts
  - Solve engine-setup limitations
  - Inter-host key enrollment
    - websocket proxy uses a similar approach

THANK YOU!

http://www.ovirt.org
fromani@redhat.com
Irc: #vdsm on #freenode #ovirt on OFTC

Backup slide: proxy on diff. host

- The ovirt-vmconsole package is already generic
- The ovirt-vmconsole-list.py helper is not making assumption on Engine position
  - Already takes full URL as target
  - Already communicates using HTTP
- The only blocker is the automated setup
  - Manual setup complex but possible
  - Documentation: http://www.ovirt.org/Serial_Console_Setup

Backup slide: how it looks

fromani@shinji

File Edit View Search Terminal Tabs Help

fromani@c7:/usr/local/ovirt-engine/bin

```
SHINji> 09:29:03 fromani [~]$ ssh -i ~/.ssh/sercon -t -p 2222 ovirt-vmconsole@192.168.1.198
Available Serial Consoles:
00 sercon_test-1[15628d2e-2735-4310-b252-ad2be1bd459f]
SELECT> 0


CentOS Linux 7 (Core)
Kernel 3.10.0-123.el7.x86_64 on an x86_64

serconvm login: root
Password:
Last login: Wed Jul 29 09:16:37 on hvc0
[root@serconvm ~]# w
 09:29:16 up 16 min,  2 users,  load average: 0.00, 0.01, 0.05
USER     TTY      LOGIN@   IDLE   JCPU   PCPU WHAT
root     tty1     09:15    1:08   0.23s  0.23s -bash
root     hvc0     09:29    4.00s  0.08s  0.02s w
[root@serconvm ~]# tty
/dev/hvc0
[root@serconvm ~]#
Broadcast message from root@serconvm.rokugan.lan (tty1) (Wed Ju

hello, serial consolers!

[root@serconvm ~]# wall 'hello back from serial console!'
[root@serconvm ~]#
```

sercon_test-1:1 - Press

File View Send key Help

```
[root@serconvm ~]# w
 09:29:20 up 16 min,  2 users,  load average: 0.00, 0.
USER     TTY      LOGIN@   IDLE   JCPU   PCPU WHAT
root     tty1     09:15    0.00s  0.24s  0.00s w
root     hvc0     09:29    8.00s  0.06s  0.06s -bash
[root@serconvm ~]# tty
/dev/tty1
[root@serconvm ~]# wall 'hello, serial consolers!'
[root@serconvm ~]#
Broadcast message from root@serconvm.rokugan.lan (tty1

hello, serial consolers!

[root@serconvm ~]#
Broadcast message from root@serconvm.rokugan.lan (hvc0

hello back from serial console!

[root@serconvm ~]# _
```