Ovirt guest agent

Date: November 2011
Speaker Name: Barak Azulay
System group manager @ RHEVM Engineering, Red Hat
Development: Gal Hammer
Agenda

- A word about guest tools
- Why guest agent?
- Matahari, virt-agent, ovirt-guest-agent
- Ovirt-guest-agent - overview
- VDSM <-> ovirt-guest-agent protocol
- Automatic login / SSO
- Current OSS status
- Roadmap
- Resources
Guest tools on RHEVM

- USB
- balloon
- Virtio-net
- Virtio-block
- Virtio-serial
- Spice agent
- SSO
- Spice driver
- guest Agent

Guest

RHEL / RHEV-H
Why guest agent

• Understand what is happening on the guest OS
  • Supply run time information (mem, users, cpu, networking...)
  • Report internal guest events that the management system would like to be aware of (User shut the guest down, user logged in ..)

• Perform various management operation
  • Quiesce,
  • Shutdown
  • ...
Ovirt-guest-agent vs virt-agent / Matahari?

- Matahari:
  - A generic purpose framework aimed for system management & monitoring.
  - Supports D-BUS & QMF (AMQP)
  - It has a specific role in the cloud
  - Still work in progress, arch over virtio-serial is evolving

- Virt-agent (qemu-ga):
  - Qemu specific – it was aimed for specific qemu needs (quiesce)
  - Communicates directly with qemu
  - Includes already various API calls – so far linux only

- Ovirt-guest-agent:
  - Exists for a long time (~5 years) – considered stable
  - Started as rhevm specific but evolved a lot since then
  - Currently the only fully functional guest agent available for ovirt
Ovirt-guest-agent

- Written in python
- Communicates with VDSM over a virtio-serial device
- Provides:
  - Information / Stats
  - Events
  - Execution of certain commands within the guest
- Supports various guest OS flavors
  - Windows XP (32)
  - Windows 7 (32/64)
  - Windows 2003 (32/64/R2)
  - Windows 2008 (32/64/R2)
  - RHEL 5.X
  - RHEL 6.X
  - Fedora 15
Ovirt-guest-agent (2)
Ovirt-guest-agent (3)
VDSM <-> ovirt-guest-agent protocol

- Communicates over virtio-serial device
- Every VM started up has

```
<controller type='virtio-serial' index='0' ports='16'/>
<channel type='unix'>
  <target type='virtio' name='com.redhat.rhevm.vdsm'/>
  <source mode='bind' path='....'/>
</channel>
```

- Ovirt-guest-agent must use the same name
  - device = /dev/virtio-ports/com.redhat.rhevm.vdsm in ovirt-guest-agent.conf for linux
  - device = \\.Global\com.redhat.rhevm.vdsm in ovi in ovirt-guest-agent.ini for windows

- The protocol is stateless
- Using JSON as a message structure
VDSM <-> ovirt-guest-agent protocol (2)

- **Information**
  - Machine name - Show the virtual machine's host name.
  - Operating system version - Show the operating system's version. Linux: this value is the kernel version. Windows: it is the Windows version name (e.g. Windows XP or Windows 7).
  - IP(v4) addresses - List of all the virtual machine's IP addresses. Only IPv4 addresses are reported.
  - Installed applications - List installed applications. Linux: application list is set using the configuration file. Windows: installed applications list is based on value read from registry.
  - Available RAM - The amount of unused physical memory. This value probably include memory like cache, or else the memory usage will always be (or near) 100% usage.
  - Logged in users - List of all logged-in users.
  - Active user - The user which currently is using the virtual machine's "physical hardware", this is more of a legacy report, The ovirt-engine uses a different logic for it nowadays

**Recently added**
- VM Disk utilization
- Internal guest network mapping (MAC, name, ipv4, ipv6)
Notifications / Events

- Power Up - Send when agent start its execution.
- Power Down - unused
- Heartbeat - Message is send every few second to notify that the agent is running. The notification includes the guest's available RAM.
- User Info - Active user was changed.
- Session Lock - Desktop was locked (Windows).
- Session Unlock - Desktop was unlocked (Windows).
- Session Logoff - A user was logged off (Windows).
- Session Logon - A user was logged on (Windows).
- Agent Uninstalled - Agent was removed from system – indication for VDSM to clear its in mem cache
VDSM <-> ovirt-guest-agent protocol (4)

- Actions (functions VDSM can execute within the guest)
  - Lock - Request locking the user's desktop.
  - Login - Perform automatic login in user's behalf.
    - Different implementation for Linux & Windows
  - Logoff - Log off the active user (currently not used by ovirt-engine)
  - Shutdown - Shut down the virtual machine.
VDSM <-> ovirt-guest-agent protocol (5)

- Examples
- `{"__name__": "heartbeat", "free-ram": "1621"}
- `{"__name__": "host-name", "name": "S-WIN7-64-SVR"}
- `{"__name__": "os-version", "version": "Win 7"}
- `{"__name__": "applications", "applications": ["RHEV-Tools 3.0.26", "RHEV-Network64 3.0.6", "RHEV-Spice-Agent64 3.0.3", "RHEV-USB 3.0.5", "RHEV-Spice64 3.0.4", "RHEV-Agent64 3.0.10", "RHEV-Serial64 3.0.5", "RHEV-Block64 3.0.8"]}`
Ovirt-guest-agent & SSO (Linux)

- The automatic login on Linux is based on three components
  - The RHEV-Agent which handle the user's credentials and work flow
  - A greeter's plug-in which allow interaction with the desktop manager.
  - A PAM module which handle the PAM's conversation.

Currently there are two greeter's plug-ins. One for GNOME desktop manager (GDM) and one for the KDE desktop manager (KDM).

- The flow:
  - The greeter's plug-in is waiting for a signal on the D-BUS interface.
  - The RHEV-Agent receive the user's credentials from the VDSM though the virtio-serial device.
  - A "User Authenticated" signal with a a one-time token is emitted by the agent. The agent also opens an abstract server socket which is used to send the user's credentials to the PAM module.
  - The plug-in starts the PAM conversation.
  - The PAM module start the conversation with a query for the token (to the plug-in).
  - The PAM module connect to the agent's abstract server socket and send the token.
  - The agent verifies the token match. And sends user's credentials to the PAM module, otherwise the connection is closed.
  - The PAM module set the down the PAM stack
ovirt-guest-agent & SSO (Linux)
Ovirt-guest-agent & SSO (Windows)

The automatic login on Windows is based on two components:

- The ovirt-guest-agent which handle the user's credentials and workflow.
- A Window's component interaction with the Winlogon system.
  - for Windows XP - the component is implemented as a GINA DLL.
  - for Win7 - The Gina interface was changed on Windows Vista with the new Credential Providers model.

Both above component will be included in the ovirt-guest-agent git repo

The flow:

- The Windows component create a named pipe and is waiting for an incoming connection.
- The RHEV-Agent receive the user's credentials from the VDSM though the virtio-serial device.
- The agent send the user's credentials though the named pipe.
- Using the user's credentials received from the named pipe, a login is performed on user's behalf.
• Current OSS status

• Git repository at:

• Contains:
  • Ovirt-guest-agent (win & linux)
  • Gdm-plugin-ovirtcred (does not compile on F15 will be soon)
  • Kdm-plugin-ovirtcred
  • Pam-ovirt-cred
  • Gina for win XP (patch sent)
  • Credentials provider for windows 7 (will be added till the end of the week)
  • Most of the work/discussions will be on vdsm lists, some will happen on the engine's lists (as needed)
• Roadmap

• Guest Agent
  • Basically features are added according to ovirt-angine's need for new features
  • Kerberos authentication (may be done through spice)
  • May evolve as a matahari plugin.

• Guest Tools
  • Creating upstream packages for guest win drivers (are there any?)
  • Decide & Create upstream about the guest tools delivery mechanism
    • Installers
    • How do the drivers get to the Vms
    • Any guest tools the community pushes
Resources

- Gerrit Wiki
- Git
  - For unregistered user
    git clone git://gerrit.ovirt.org/ovirt-guest-agent
  - For registered user
    git clone gerrit.ovirt.org:ovirt-guest-agent
- Ovirt-guest-agent wiki
THANK YOU!

http://www.ovirt.org