Improving client systems security with Qubes OS

Marek Marczykowski-Górecki, Invisible Things Lab

4 Jul 2016

We need secure client systems

We need secure client systems

Otherwise no security really works:

We need secure client systems

Otherwise no security really works:

- Encryption
- 2-factor authentication
- "Secure cloud"

All the above can be manipulated by compromised client system.

Current, monolithic systems

- All the drivers, services part of TCB

Current, monolithic systems

- All the drivers, services part of TCB
  - Networking - DHCP client, Wifi

Current, monolithic systems

- All the drivers, services part of TCB
  - Networking - DHCP client, Wifi
- Every application have access to all the user data

Current, monolithic systems

- All the drivers, services part of TCB
  - Networking - DHCP client, Wifi
- Every application have access to all the user data
  - PDF viewer

Current, monolithic systems

- All the drivers, services part of TCB
  - Networking - DHCP client, Wifi
- Every application have access to all the user data
  - PDF viewer
  - Web browser

Current, monolithic systems

- All the drivers, services part of TCB
  - Networking - DHCP client, Wifi
- Every application have access to all the user data
  - PDF viewer
  - Web browser
  - Mail client

Current, monolithic systems

- All the drivers, services part of TCB
  - Networking - DHCP client, Wifi
- Every application have access to all the user data
  - PDF viewer
  - Web browser
  - Mail client
- Lack of GUI separation

Current, monolithic systems

- All the drivers, services part of TCB
  - Networking - DHCP client, Wifi
- Every application have access to all the user data
  - PDF viewer
  - Web browser
  - Mail client
- Lack of GUI separation
- One bug to rule them all

Compartmentalization

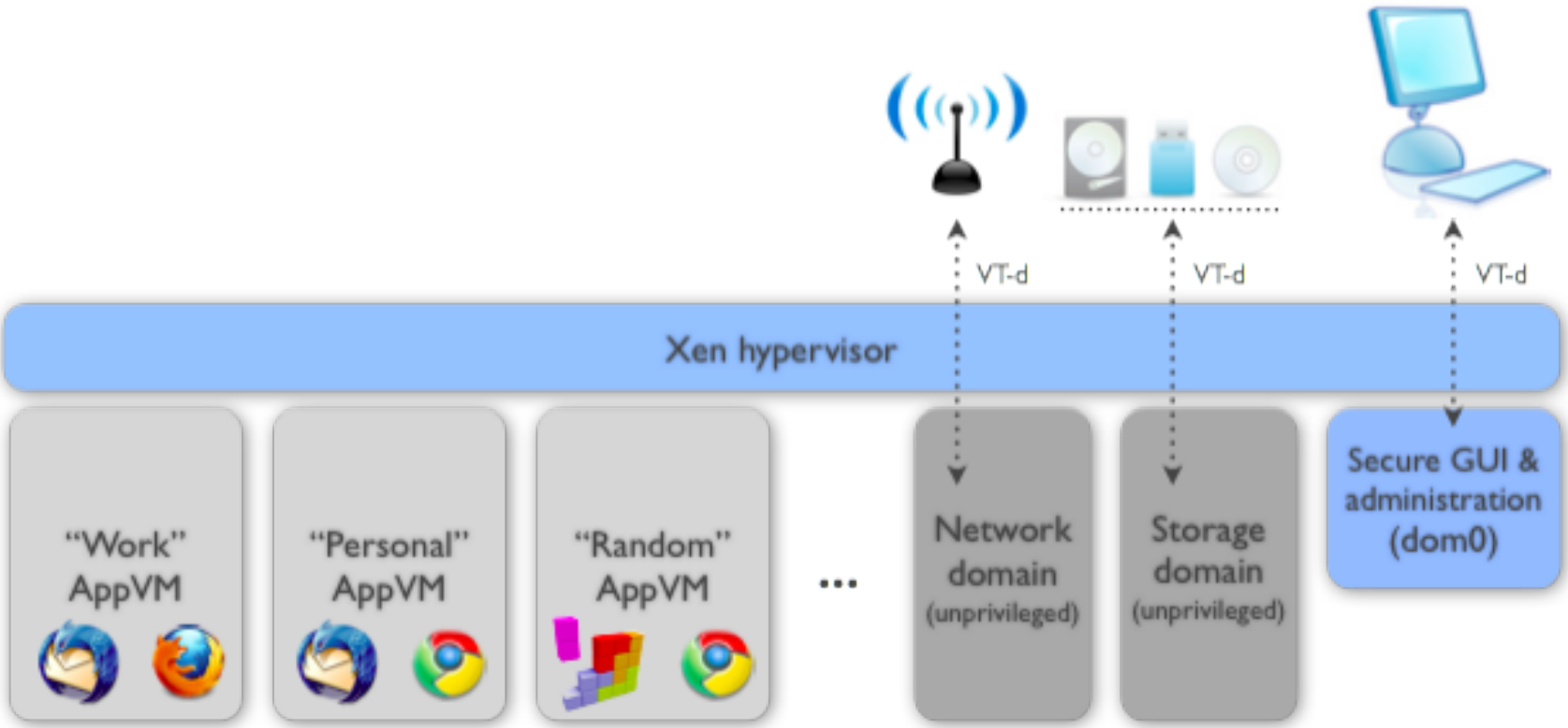
- Separate each component from each other

Compartmentalization

- Separate each component from each other
- Divide the whole system into security domains

Compartmentalization

- Separate each component from each other
- Divide the whole system into security domains
- Use virtualisation, have **minimal, simple** interfaces


VT-d
VT-d
VT-d
Xen hypervisor
"Work" AppVM
"Personal" AppVM
"Random" AppVM
...
Network domain (unprivileged)
Storage domain (unprivileged)
Secure GUI & administration (dom0)

Isolated devices

- Net VM

Isolated devices

- Net VM
- USB VM

Isolated devices

- Net VM
- USB VM
- GUI VM (planned)

Isolated data

- Different areas of digital life
- Different levels of trust

Not only about separating applications!

Isolated workflows

- Proxy VM (VPN, Tor, Whonix)
- Data converters (PDF, images)
- Data storage (offline vaults, gpg)
- Disposable VM

Framework for building secure workflows

- Socket-like inter-VM communication (qrexec)
- Each operation requires policy approval (not necessary user approval)
- VMs - building blocks

Block/USB devices handling

- Dedicated USB VM

Block/USB devices handling

- Dedicated USB VM
- Untrusted, no user data

Block/USB devices handling

- Dedicated USB VM
- Untrusted, no user data
- Services for specific applications

Block/USB devices handling

- Dedicated USB VM
- Untrusted, no user data
- Services for specific applications

Block/USB devices handling

- Dedicated USB VM
- Untrusted, no user data
- Services for specific applications

Supported devices

- Block devices - can be attached to any VM

Block/USB devices handling

- Dedicated USB VM
- Untrusted, no user data
- Services for specific applications

Supported devices

- Block devices - can be attached to any VM
- Input devices - strictly filtered (mouse only/mouse+keyboard)

Block/USB devices handling

- Dedicated USB VM
- Untrusted, no user data
- Services for specific applications

Supported devices

- Block devices - can be attached to any VM
- Input devices - strictly filtered (mouse only/mouse+keyboard)
- Generic USB passthrough

Split GPG

- Like software-based smartcard

Split GPG

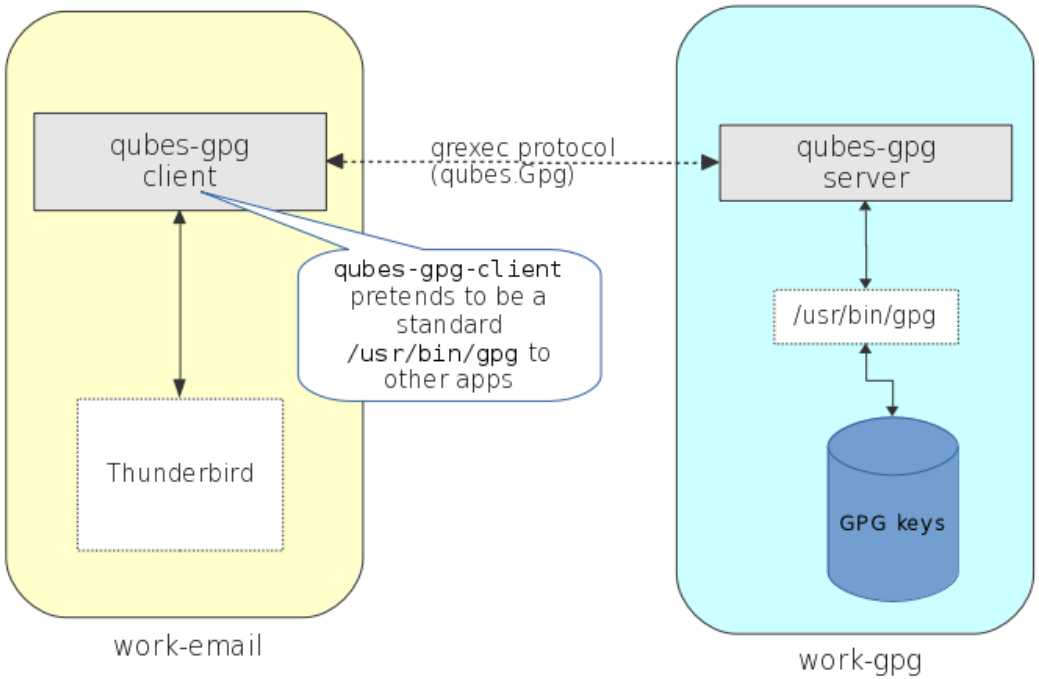
- Like software-based smartcard
- Better control

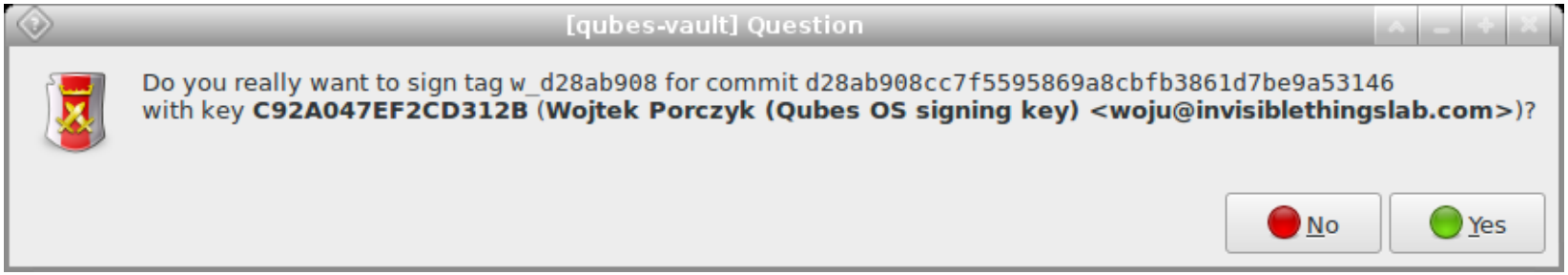
Split GPG

- Like software-based smartcard
- Better control
- Audit trail

Split GPG

- Like software-based smartcard
- Better control
- Audit trail
- /bin/gpg drop-in replacement


qubes-gpg
client
grexec protocol
(qubes.Gpg)
qubes-gpg
server
qubes-gpg-client
pretends to be a
standard
/usr/bin/gpg to
other apps
/usr/bin/gpg
Thunderbird
GPG keys
work-email
work-gpg


[qubes-vault] Question
Do you really want to sign tag w_d28ab908 for commit d28ab908cc7f5595869a8cbfb3861d7be9a53146
with key C92A047EF2CD312B (Wojtek Porczyk (Qubes OS signing key) <woju@invisiblethingslab.com>)?
No
Yes

Networking

- Network devices in separate VMs

Networking

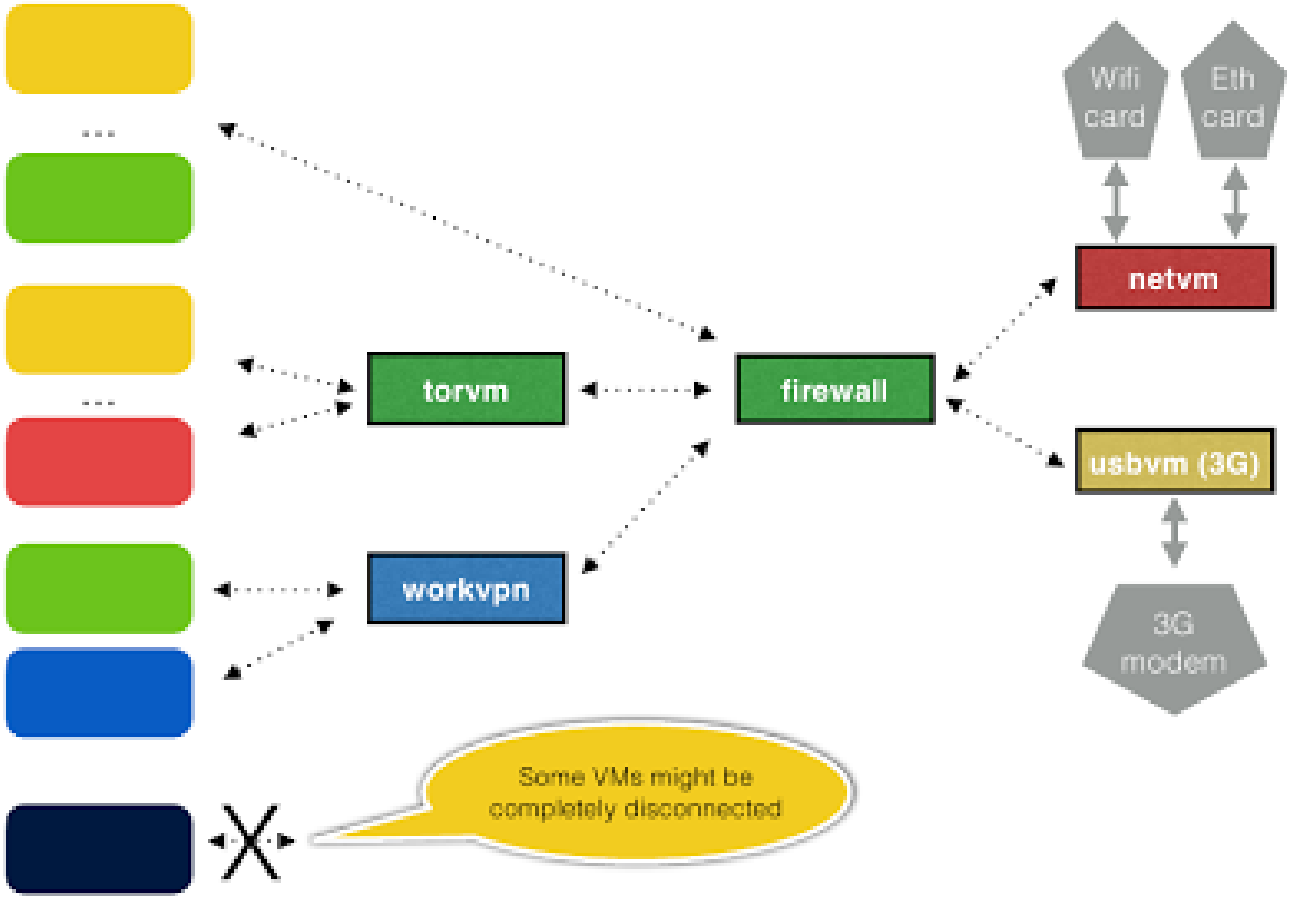
- Network devices in separate VMs
- Dom0 has no network at all!

Networking

- Network devices in separate VMs
- Dom0 has no network at all!
- VMs can be chained (proxy)

Networking

- Network devices in separate VMs
- Dom0 has no network at all!
- VMs can be chained (proxy)
- VPN, Tor, IDS, and many more


...
...
torvm
workvpn
firewall
Wifi card
Eth card
netvm
usbvm (3G)
3G modem
Some VMs might be completely disconnected

Qubes 3.x

- Qubes 3.1 released in March 2016
- Qubes 3.2-rc1 released last month

Supports everything mentioned here.

Qubes 3.x

- Qubes 3.1 released in March 2016
- Qubes 3.2-rc1 released last month

Supports everything mentioned here.

Management stack

- Powerful framework for building secure client system.
- Sample recipes provided
- We need more!

Qubes Master Signing Key

427F 11FD 0FAA 4B08 0123 F01C DDFA 1A3E 3687 9494

Questions?

Thanks!