

SELinux from Developer POV

LinuxDays 2017

Lukas Vrabc
Vit Mojzis

Virtual machine setup

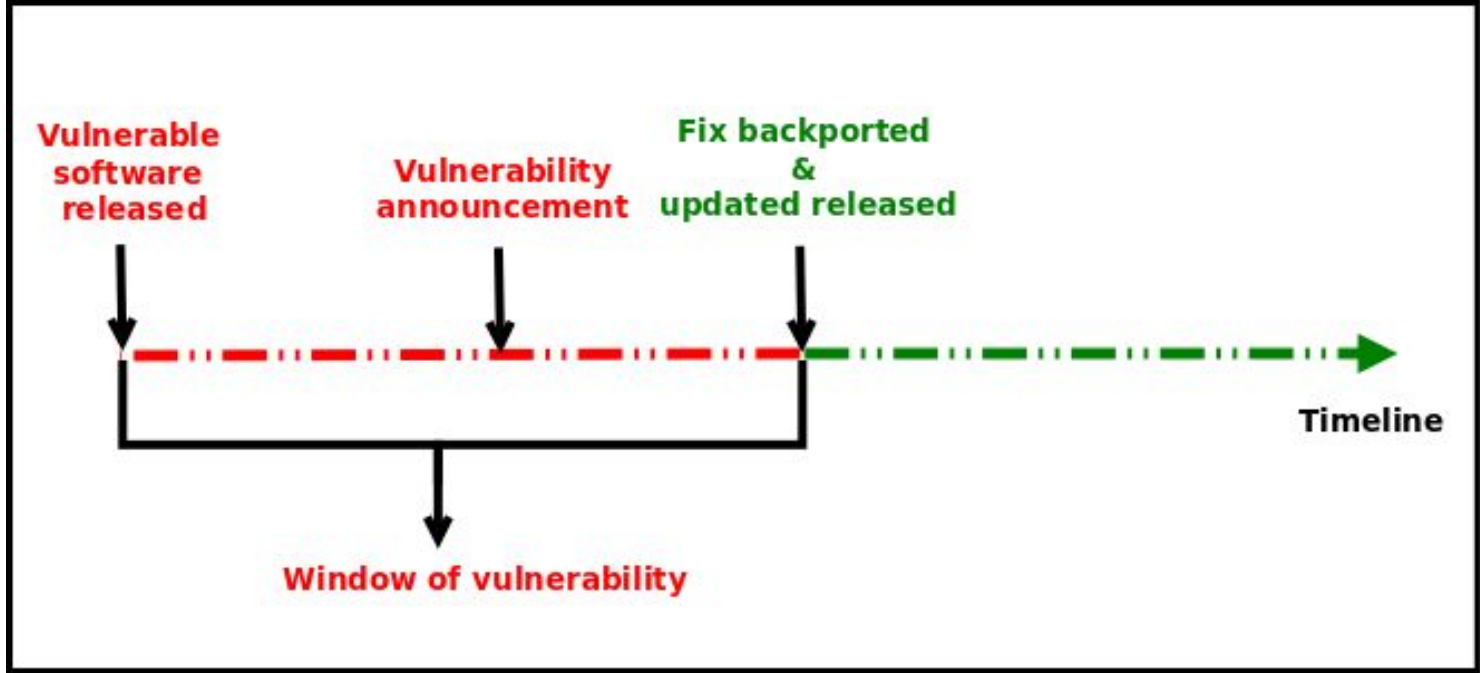
- http://lukas-vrabec.com/image_selinux.tar.xz
- Run “virtual machine manager”
- Create new virtual machine
 - Import existing image
 - Os Type - Linux
 - Customize configuration before install
 - Add hardware
 - Storage, CDROM, cloudinit_iso

Agenda

- Why SELinux ?
- Why ship your own SELinux module ?
- How can I add custom SELinux module into project rpms?
- How can I create Fedora module with custom SELinux module?

Why SELinux?

REACTIVE SECURITY



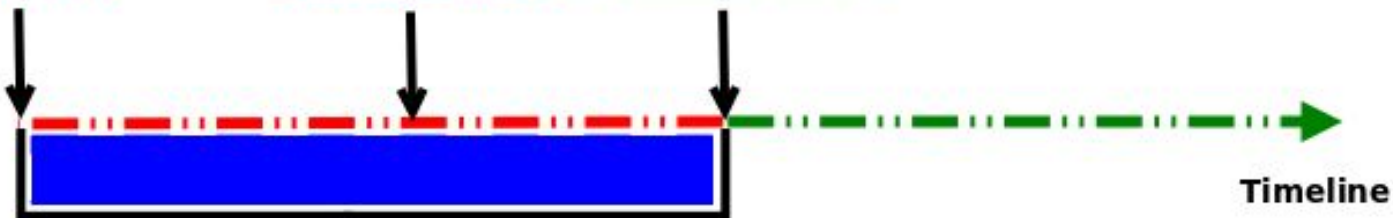
YOUR SYSTEM **IS NOT PROTECTED** DURING THE
WINDOW OF VULNERABILITY!

PROACTIVE SECURITY

Vulnerable software released

Vulnerability announcement

Fix backported & updated released



Window of vulnerability is filled by proactive security

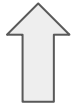
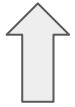
PROACTIVE SECURITY HELPS TO **PROTECT** YOUR
SYSTEM DURING THE WINDOW OF VULNERABILITY!

SECURITY ENHANCED LINUX IS A SECURITY
MECHANISM BRINGING PROACTIVE SECURITY FOR
YOUR SYSTEM.

Traditional Linux Security

```
$ ls -dl /var/www/html/
```

```
drwx r-x r-x. 2 root root /var/www/html/
```



USER GROUP ALL

SELinux Security Policy

CORE COMPONENT OF SELINUX

CORE COMPONENT OF SELINUX
COLLECTION OF SELINUX POLICY RULES

CORE COMPONENT OF SELINUX
COLLECTION OF SELINUX POLICY RULES
**LOADED INTO THE KERNEL BY SELINUX
USERSPACE TOOLS**

ENFORCED BY THE KERNEL

ENFORCED BY THE KERNEL

**USED TO AUTHORIZE ACCESS REQUESTS ON THE
SYSTEM**

BY DEFAULT **EVERYTHING** IS DENIED AND YOU
DEFINE POLICY RULES TO ALLOW CERTAIN
REQUESTS.

SELINUX POLICY RULES

DESCRIBE AN **INTERACTION** BETWEEN PROCESSES
AND SYSTEM RESOURCES

SELINUX VIEW OF THAT INTERACTION

```
ALLOW apache_process apache_log:FILE  
      READ;
```


apache_process apache_log

ARE LABELS

LABELS

ASSIGNED TO PROCESSES

ASSIGNED TO PROCESSES

ASSIGNED TO SYSTEM RESOURCES

ASSIGNED TO PROCESSES
ASSIGNED TO SYSTEM RESOURCES
BY SELINUX SECURITY POLICY

ASSIGNED TO PROCESSES
ASSIGNED TO SYSTEM RESOURCES
BY SELINUX SECURITY POLICY

**MAP REAL SYSTEM ENTITIES INTO THE SELINUX
WORLD**

LABELS IN REALITY

STORED IN EXTENDED ATTRIBUTES OF FILE
SYSTEMS - EXT2,EXT3, EXT4 ...


```
# getfattr -n security.selinux /etc/passwd
getfattr: Removing leading '/' from absolute path
names
# file: etc/passwd
security.selinux="system_u:object_r:passwd_file_t:s0"

# ls -Z /etc/passwd
system_u:object_r:passwd_file_t:s0 /etc/passwd
```

Benefits of shipping own SELinux module

- Changes in a policy can be modified immediately, so the product package maintainer does not need to wait until the distribution SELinux policy is updated.
- Policy changes in product SELinux policy can be released together with changes in product package so SELinux policy will be always synchronized with a product.
- Product package can follow different timeline deadlines then SELinux policy package, this can cause issues and customer can get new product package version without necessary changes in SELinux policy and this can block some functionality of a product.

https://fedoraproject.org/wiki/SELinux/IndependentPolicy#Creating_Own_Product_Policies

Independent SELinux policy module

- Write own SELinux policy from scratch and ask SELinux team for policy review. Note that a guide how to write an SELinux policy from the scratch is not a part of this workshop (See the [Generating SELinux Policy Modules: sepolicy generate](#) section in the SELinux Guide).
- Extract an SELinux policy from a distribution policy package. The Git repository with distribution policies is located on github.com/fedora-selinux/selinux-policy and github.com/fedora-selinux/selinux-policy-contrib.

Agreement workflow

Before you start with shipping own product policies, let the Red Hat SELinux team know about your intentions. To do this, use Fedora mailing list or contact SELinux policy maintainer:

- SELinux Policy maintainer
- selinux@lists.fedoraproject.org

Git Repository setup

Create directory to contain the project

\$ mkdir myapp-selinux

\$ cd myapp-selinux

initialize git repository

\$ git init

Push git repository to remote e.g. to github.com

\$ git remote add origin git@github.com:username/myapp-selinux

\$ git push -u origin master

Preparing sources for the Policy Git Repository

- **License**

- A Git repository should not contain only SELinux policy source files, but also a license. For more information how to add an open source license in your repository, see the [Adding a license to a repository](#) article on the GitHub Help. Distribution policies have GPL license, so any policy extracted from Distribution policy must have GPL compatible license.

- **Makefile**

- https://fedoraproject.org/wiki/SELinux/IndependentPolicy#Creating_Own_Product_Policies
- In section Makefile

- **Policy source**

- Type enforcement file (*.te)
- File contexts file (*.fc)
- Interface file (*.if)

\$ ls

Makefile myapp.fc myapp.if myapp.te LICENSE

\$ make

make -f /usr/share/selinux/devel/Makefile myapp.pp

make[1]: Entering directory '/home/lvrabec/devel/documentations/examples'

Compiling targeted myapp module

/usr/bin/checkmodule: loading policy configuration from tmp/myapp.tmp

/usr/bin/checkmodule: policy configuration loaded

/usr/bin/checkmodule: writing binary representation (version 17) to tmp/myapp.mod

Creating targeted myapp.pp policy package

rm tmp/myapp.mod.fc tmp/myapp.mod

make[1]: Leaving directory '/home/lvrabec/devel/documentations/examples'

Compressing myapp.pp -> myapp.pp.bz2

bzip2 -9 myapp.pp

```
$ cd ../
```

```
$ tar -czf myapp-selinux.tar.gz myapp-selinux/
```

SELinux policy is ready!

Creating spec file

Spec file will be described on the Independent Policy wiki page:

https://fedoraproject.org/wiki/SELinux/IndependentPolicy#Creating_Own_Product_Policies

Setting booleans During a package installation

Usage of booleans in a .spec file follows these rules:

- If a boolean mentioned in the product .spec file is not set by user previously, it will be changed in the %post install phase and during the %post uninstall phase will be reverted.
- If a boolean mentioned in the product .spec file was set by user previously, it will be changed to a value from this file. However, during the uninstallation of a product SELinux subpackage, it will not be reverted.

Port labelling during a
package installation

```
if %{_sbindir}/selinuxenabled ; then
    %{_sbindir}/load_policy
    %relabel_files
    %{_sbindir}/semanage port -a -t product_port_t -p tcp 1111
fi
```

Move your SELinux product policy sources to the proper destination:

```
$ cp myapp-selinux.tar.gz ~/rpmbuild/SOURCES/
```

Build your product (sub)package with an own SELinux policy:

```
# rpmbuild -ba myapp-selinux.spec
```

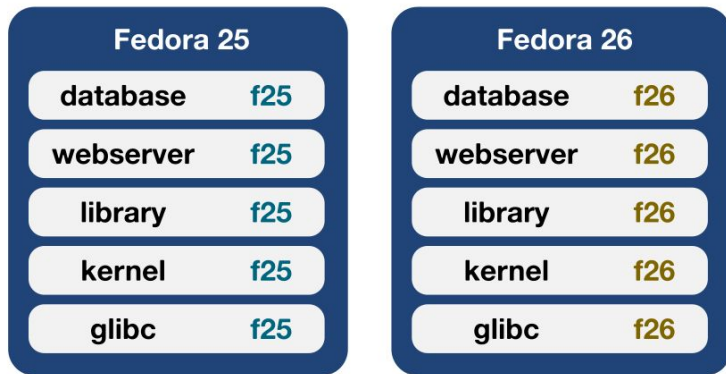
Removing an Own Product Policy from the System Policy

When is your own product SELinux subpackage ready for a release, contact the SELinux policy maintainer. He should remove a product policy from the SELinux distribution policy and update the package. A product maintainer should add dependency for the selinux-policy package:

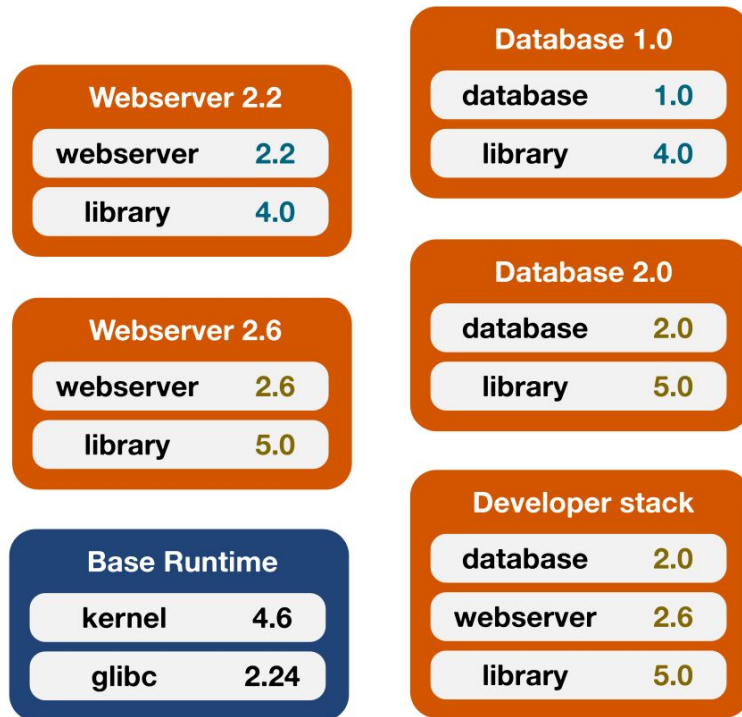
```
# Version of selinux-policy when product policy was removed
%global selinux_policyver POLICY_VERSION
Requires: selinux-policy >= %{selinux_policyver}
```


How can I create Fedora
module with custom
SELinux module?

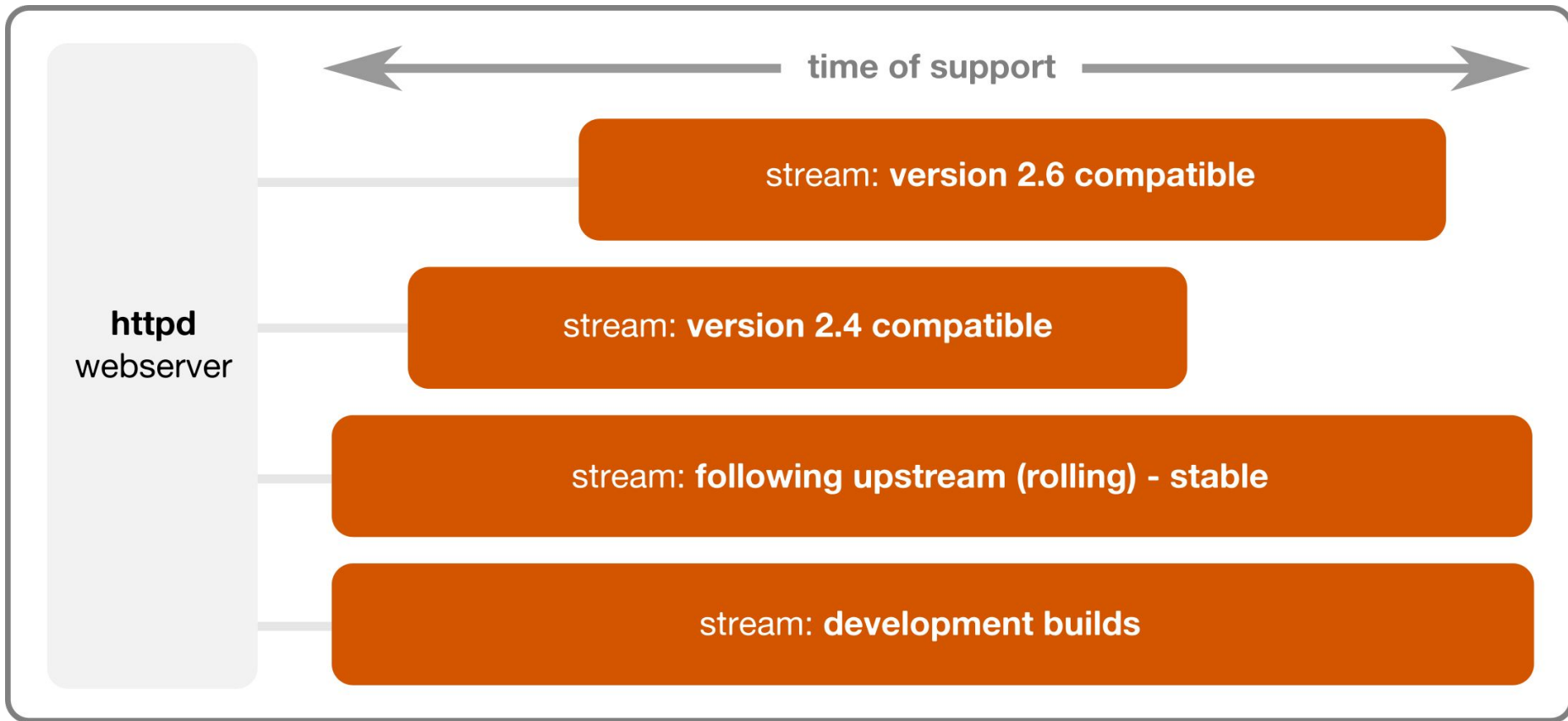
Traditional Distribution



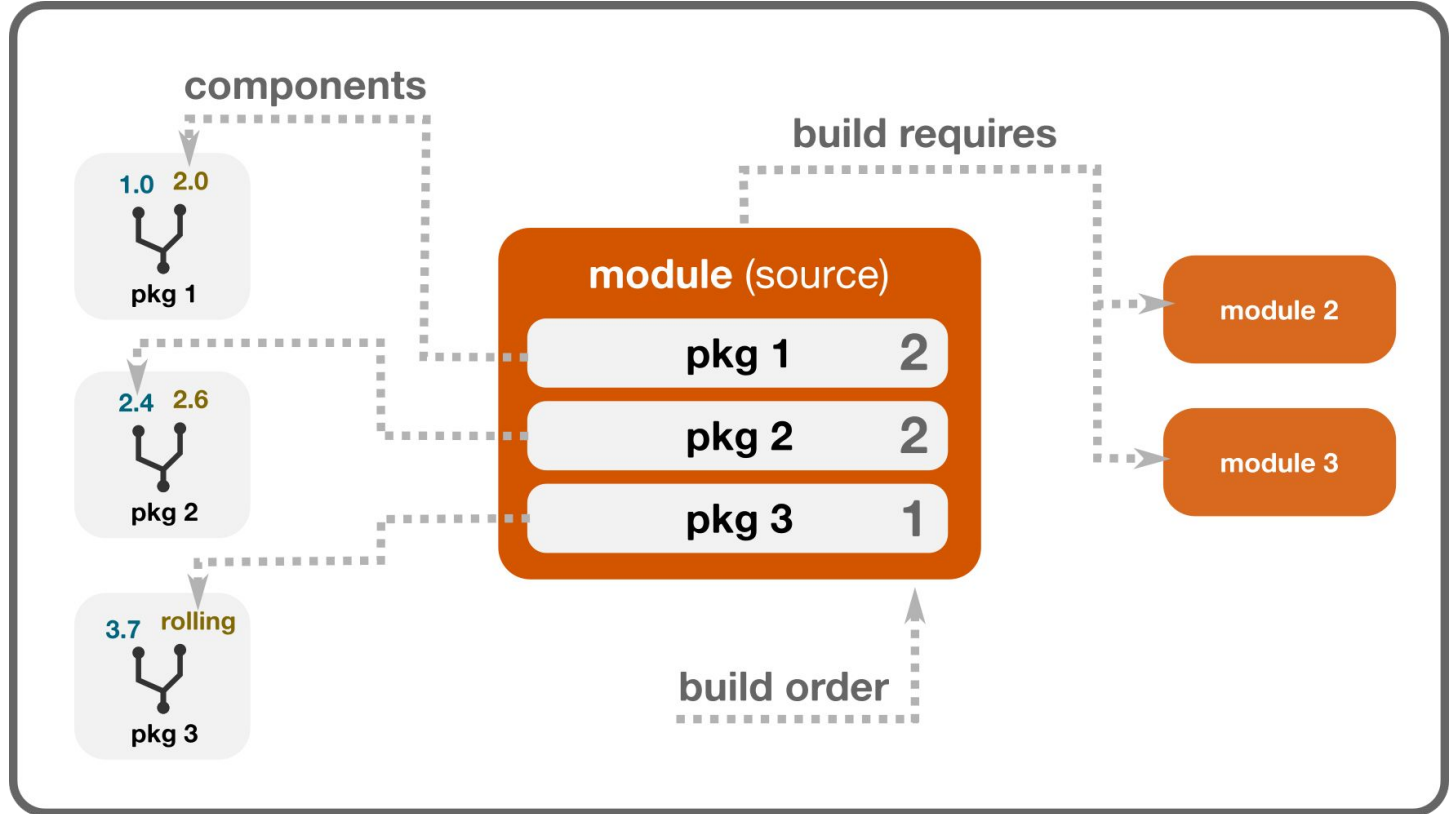
Modularity



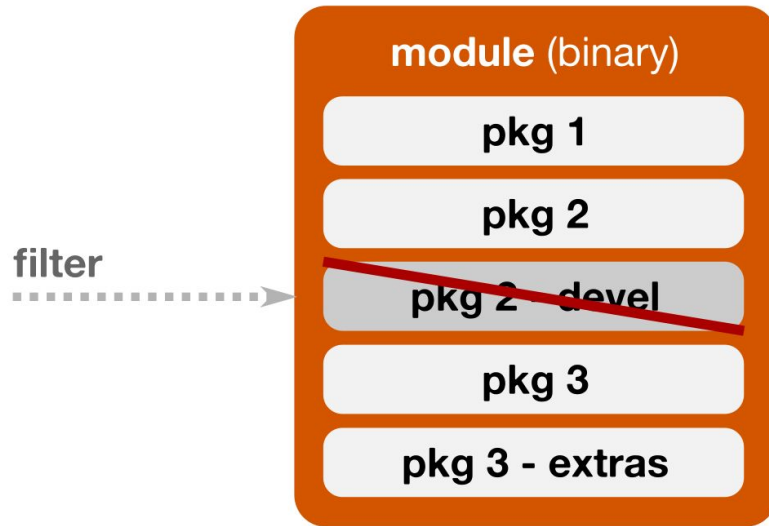
Module streams



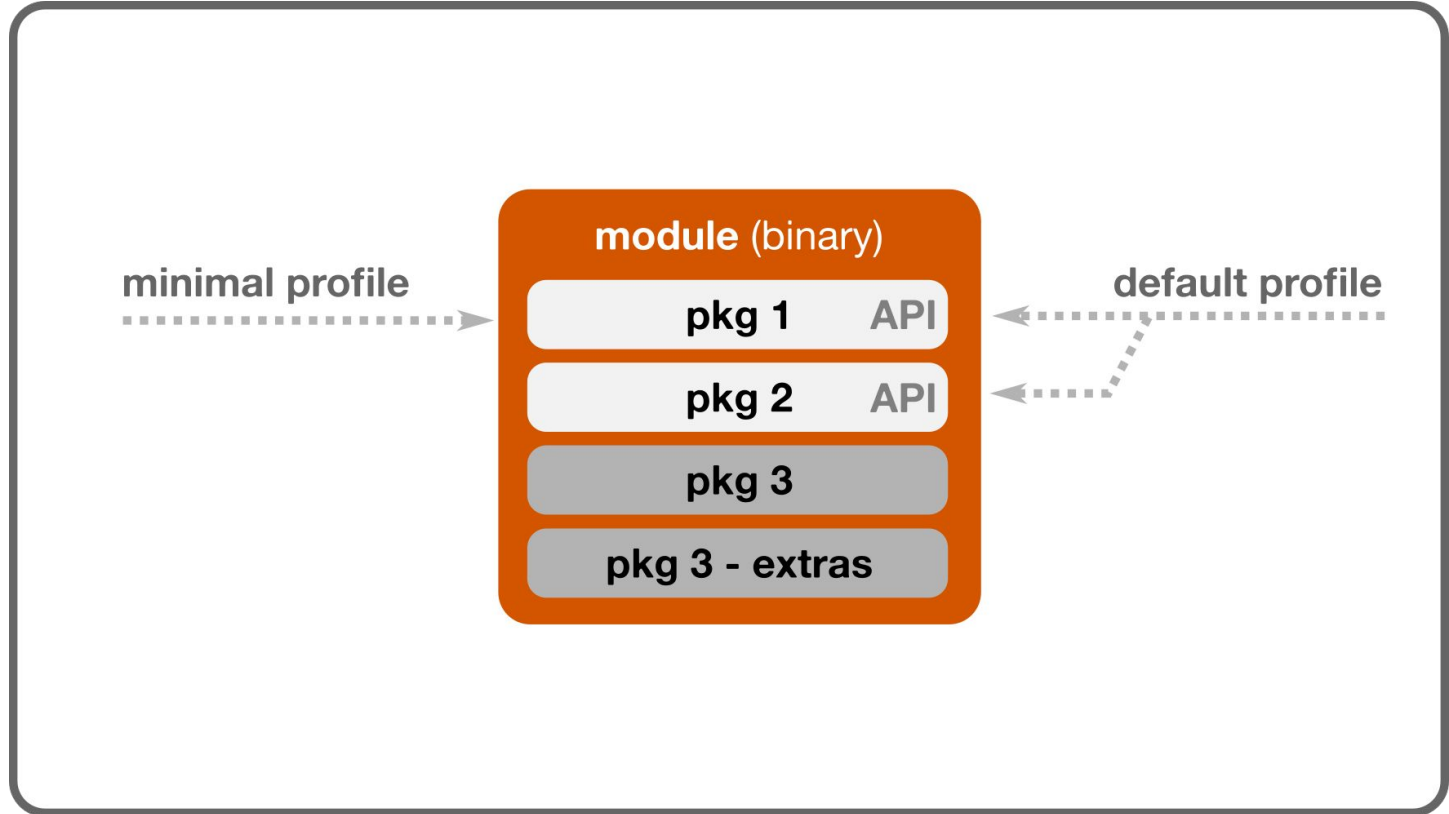
Define how to build the module



Decide what to ship



Specify how to use



QUESTIONS?

Miroslav Grepl's blog <https://mgrepl.wordpress.com/>
Paul Moore's blog <http://www.paul-moore.com/>
Lukas Vrabec's blog <https://lukas-vrabec.com/>
Dan Walsh's blog <http://danwalsh.livejournal.com/>

THANK YOU