



The Linux & Open Source Company

State of deb-support in Katello

Where we are, where we go and where we want to be

Markus Bucher, Quirin Pamp

- ▶ Software Developer @ATIX-AG
- ▶ located near Munich, Germany
- ▶ coding and using Linux since last millenium
- ▶ coding in Foreman & Co. since 2018
- ▶ not active on Social Media*
- ▶  m-bucher
- ▶ @mebuh:matrix.org
- ▶ bucher@atix.de



orcharhino 
Automate your Infrastructure - Free your business

- 1 Where we are
- 2 Version Compare in DB
- 3 Structured APT Repository Publishing
- 4 Debian/Ubuntu Errata
 - How it works now
 - How it could also work
 - How do we get this upstream
- 5 Sneak Peaks

What happened last year

- ▶ #35713 Add Debian Content to new Host Details thanks @nadjaheitmann
- ▶ #38096 Fix flat APT repo handling thanks @quba42
- ▶ #36981 Upload Structured Debian Content thanks @Manisha15
- ▶ #36764 Export and Import Debian content thanks @hstct
- ▶ #37794 Show more Deb-Package Details in UI thanks @Thorben-D
- ▶ #31257 Incremental update with Deb-Packages (hammer-cli only)
- ▶ #38083 Repository drop-down selector on Debian packages index page
- ▶ #. Template improvements thanks @sbernhard et al.

various smaller improvements and bugfix
thanks @everyone for reporting, fixing, and reviewing

Also



we got merge permissions for Katello

- 1 Where we are
- 2 Version Compare in DB
- 3 Structured APT Repository Publishing
- 4 Debian/Ubuntu Errata
 - How it works now
 - How it could also work
 - How do we get this upstream
- 5 Sneak Peaks

Why do we need this

- ▶ **Filters:** latest 3 Versions of a package
- ▶ **UI:** show only the latest Version of packages to select for installation
- ▶ **Applicability:** Host has Version A, Repo has Version B. Is package applicable?
- ▶ **Errata:** Erratum requires at least Version X. Is it applicable/installable?

The screenshot shows the Foreman web interface. The top navigation bar includes the Foreman logo, 'Default Organization', 'Default Location', and 'Admin User'. A sidebar on the left contains navigation options like 'Monitor', 'Content', 'Subscriptions', etc. The main content area displays details for the host 'debian12.example.com'. Under the 'Content' tab, the 'Packages' sub-tab is active, showing a table with one package: 'rsync'. The package is in an 'Upgradable' state, with an installed version of '3.2.7-1' and an upgrade target of '3.2.7-1+deb12u2'. The interface includes search bars, filters, and pagination controls.

Package	Status	Installed version	Upgradable to
rsync	↑ Upgradable	3.2.7-1	3.2.7-1+deb12u2

The Problem Challenge

1	=	0:1	Epoch defaults to 0
1.09	=	1.9	Leading 0 are ignored
1.0	<	1.1	1 is larger than 0
1.2	>	1.2~rc-3	Tilde is always smaller
1.2.0~b7-1	>	1.2.0~b6-1	Right-side is eval'ed as usual

Feel free to try it

```
1 dpkg --compare-versions '1.0' '<<' '1.1' && echo 'YES'
```


It's hard ;-)

PostgreSQL does **not** support Debian version compare out-of-the-box

- ▶ native extension exists (adds column-type → supports Indexing)

<https://salsa.debian.org/postgresql/postgresql-debversion>

can't use in foreman-ecosystem :-(

Currently: PL/pgSQL compare function `deb_version_cmp()`

```
1 > SELECT deb_version_cmp('1.1', '1.0'); --> 1
```

- ▶ implements compare in PL/pgSQL ([🔗 ATIX-AG/postgresql-debversion-evr](https://salsa.debian.org/postgresql/postgresql-debversion-evr))
- ▶ is slower (50k compares ~3s)
- ▶ can't be used for indexing => no (efficient) sorting

- 1 Where we are
- 2 Version Compare in DB
- 3 Structured APT Repository Publishing
- 4 Debian/Ubuntu Errata
 - How it works now
 - How it could also work
 - How do we get this upstream
- 5 Sneak Peaks

Motivation

- ▶ APT repositories have structure to supply packages of multiple
 - ▶ releases with various
 - ▶ components in diverse
 - ▶ architectures
- ▶ provide (deb-)content to hosts (and others)
 1. hosts just take all packages within a ContentView-Version
 2. others may also need repo meta-information, e.g.
 - ▶ release-name, etc. for APT-pinning or Unattended Upgrades
 - ▶ architecture, components for selecting packages

pulp was configured to do 'simple'-publishing

=> put all packages into release 'default' and component 'all'

Solution (in pulp)

- ▶ use structured Publisher in Pulp
- ▶ requires structure information in Pulp repository saved associations:
 - ▶ `package - release - components`
 - ▶ `package - release - architectures`
 - ▶ ...

new problem:

- ▶ Katello must inform host how to configure repo:
 - ▶ `release`
 - ▶ `components`
 - ▶ `architectures`

Solution Katello (and beyond)

host needs release, components, **and** architectures

- ▶ transfer via Candlepin as part of repo-url:

`https://[...]/?comp=updates/main&rel=bookworm-security`

- ▶ Subscription-Manager configures APT

- ▶  candlepin/subscription-manager #3454

- ▶ deb-packages: <https://oss.atix.de/>

```
1 name: Debian 12 security amd64 main
2 baseurl: https://[...]/Debian_12_security_amd64_main/%3Fcomp%3Dupdates/main%26rel%3Dbookworm-security
3 arches: amd64
4 Types: deb
5 URIs: katello://[...]/custom/Debian_12_security/Debian_12_security_amd64_main/
6 Suites: bookworm-security
7 Components: updates/main
8 Trusted: yes
9 Architectures: amd64
10 Signed-By: /etc/apt/trusted.gpg.d/orcharhino_key.asc
11 id: Default_Organization_Debian_12_security_Debian_12_security_amd64_main
```

Everything is Awesome!?

structure information may change over time:

- ▶ components are added
- ▶ release names may change
- ▶ ...

this changes the content-URL!

→ content now belongs to Repository(-Version) instead of Root-Repository.

Awesome!

- 1 Where we are
- 2 Version Compare in DB
- 3 Structured APT Repository Publishing
- 4 Debian/Ubuntu Errata
 - How it works now
 - How it could also work
 - How do we get this upstream
- 5 Sneak Peaks

- 1 Where we are
- 2 Version Compare in DB
- 3 Structured APT Repository Publishing
- 4 Debian/Ubuntu Errata
 - How it works now
 - How it could also work
 - How do we get this upstream
- 5 Sneak Peaks

Data source

- ▶ **Debian:** List of DSA (DebianSecurityAnnouncement) using data from security.debian.org
other sources:
 - ▶ OVAL
 - ▶ MailingList
- ▶ **Ubuntu:** List of USN (UbuntuSecurityNotice) `database.json` including all USN of all versions
other sources:
 - ▶ REST-API to request specific list of USN

dedicated Parser/Server preparing information

📦 ATIX-AG/errata_parser 📦 ATIX-AG/errata_parser

- ▶ always recreates all errata
 - ▶ should only create errata for new security announcements
- ▶ includes all available binary packages for security announcement
 - ▶ can be quiet a lot (e.g. `linux` on Ubuntu ESM)
 - ▶ takes current state of security-repositories (moving target)

```
1 {
2   "name": "DSA-5850-1",
3   "title": "git -- security update",
4   "issued": "26 Jan 2025",
5   "affected_source_package": "git",
6   "packages": [
7     {
8       "name": "git",
9       "version": "1:2.39.5-0+deb12u2",
10      "architecture": "amd64",
11      "component": "main",
12      "release": "bookworm"
13    },
14    ...
15  ],
16  "description": "Git is a fast, scalable, distributed revision control system ...",
17  "cves": [
18    "CVE-2024-50349",
19    "CVE-2024-52006"
20  ],
21  "severity": "not yet assigned",
22  "scope": "local",
23  "dbts_bugs": [
24    1093042,
25    1093042
26  ]
27 }
```

- ▶ configure address of dedicated errata server in repository
- ▶ Katello requests errata JSON for release (e.g. bookworm-security)
- ▶ Debian Errata are handled by Katello only (no pulp)

Debian 12 security amd64 main Select Action ▾

Products > Debian 12 > Repositories > Debian 12 security amd64 main

Basic Information

Name: Debian 12 security amd64 main ✎ ✎

Label: Debian_12_security_amd64_main ✎ ✎

Description: ✎ ✎

Backend Identifier: d1c79077-2ecd-4a35-98f1-c3957e02e6d2

Type: deb

Sync Settings

Upstream URL: <http://security.debian.org/debian-security/> ✎ ✎

Releases/ Distributions: bookworm-security ✎ ✎

Components: main ✎ ✎

Architectures: amd64 ✎ ✎

Errata URL: <https://dep.atix.de/dep/api/v1/debian> ✎ ✎

Verify SSL: Yes ✎ ✎

Upstream Authorization: ✎ ✎ ✘

Content Counts

Content Type	
deb Packages	1911
Errata	382

Upload Package

If you want to upload individual packages, create a separate repository with an empty "Upstream URL" field.

Agenda



- 1 Where we are
- 2 Version Compare in DB
- 3 Structured APT Repository Publishing
- 4 Debian/Ubuntu Errata
 - How it works now
 - How it could also work
 - How do we get this upstream
- 5 Sneak Peaks

Possible Improvements

- ▶ Stateful Parser
 - ▶ do not recreate existing errata
- ▶ more Scalable/Resourceful Sources
 - ▶ e.g. Ubuntu: REST-API; Debian: OVAL
- ▶ Sync announcements directly from Debian/Ubuntu to Katello
 - ▶ obsolete central server
 - ▶ handlers for each distribution service and format in Katello
- ▶ link with available package-lists from pulp (requires source-package's name)

```
1 pulp deb content -t package list --source 'git' --repository-version '/pulp/...'
```

- ▶ no detailed pkg-info in errata transfer needed

- 1 Where we are
- 2 Version Compare in DB
- 3 Structured APT Repository Publishing
- 4 Debian/Ubuntu Errata
 - How it works now
 - How it could also work
 - How do we get this upstream
- 5 Sneak Peaks

How do we get this upstream

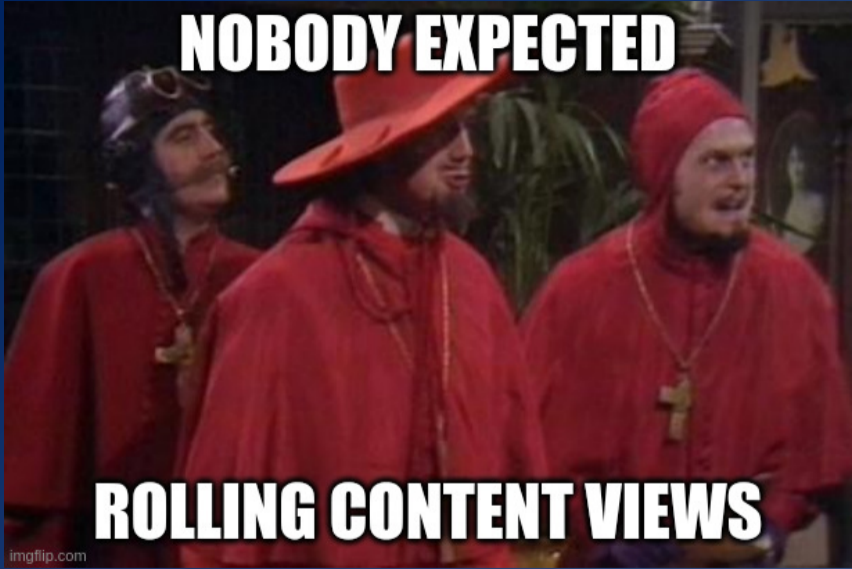


Let's discuss now and/or on Wednesday

- 1 Where we are
- 2 Version Compare in DB
- 3 Structured APT Repository Publishing
- 4 Debian/Ubuntu Errata
 - How it works now
 - How it could also work
 - How do we get this upstream
- 5 Sneak Peaks

what we also have in mind:

- ▶ `foreman-installer-module` for creating Debian Signing Key
- ▶ install from synced Debian-/Ubuntu-content
- ▶ ...
- ▶ get the stuff we do merged faster
 - ▶ get a grip on that we are now allowed to do so






imgflip.com

Rolling Content Views

- ▶ Simple Content Access (SCA) removed 'subscription' from Activation Key
 - ▶ one less thing to forget → Awesome!
 - ▶ Activation Key from Default Content View now always provides all repos
 - ▶ usually not a problem (e.g. RedHat, AlmaLinux)
 - ▶ until the 'right' customer uses it:
zypper (SUSE) shows all repos (not just enabled ones)

new Content View type: Rolling

Type		
 Content view Contains repositories. Versions are published and optionally filtered.	 Composite content view Contains content views. You must choose the version to use for each content view.	 Rolling content view Contains repositories. Always serves the latest synced content, without the need to publish versions.

main takeaways

- ▶ overall good state (right?)
- ▶ faster with changes (based on limited statistics)
- ▶ some stuff just takes time
- ▶ there are limitations (AFAIK)