



# Insights into Checkmk Development






**Lars Michelsen**

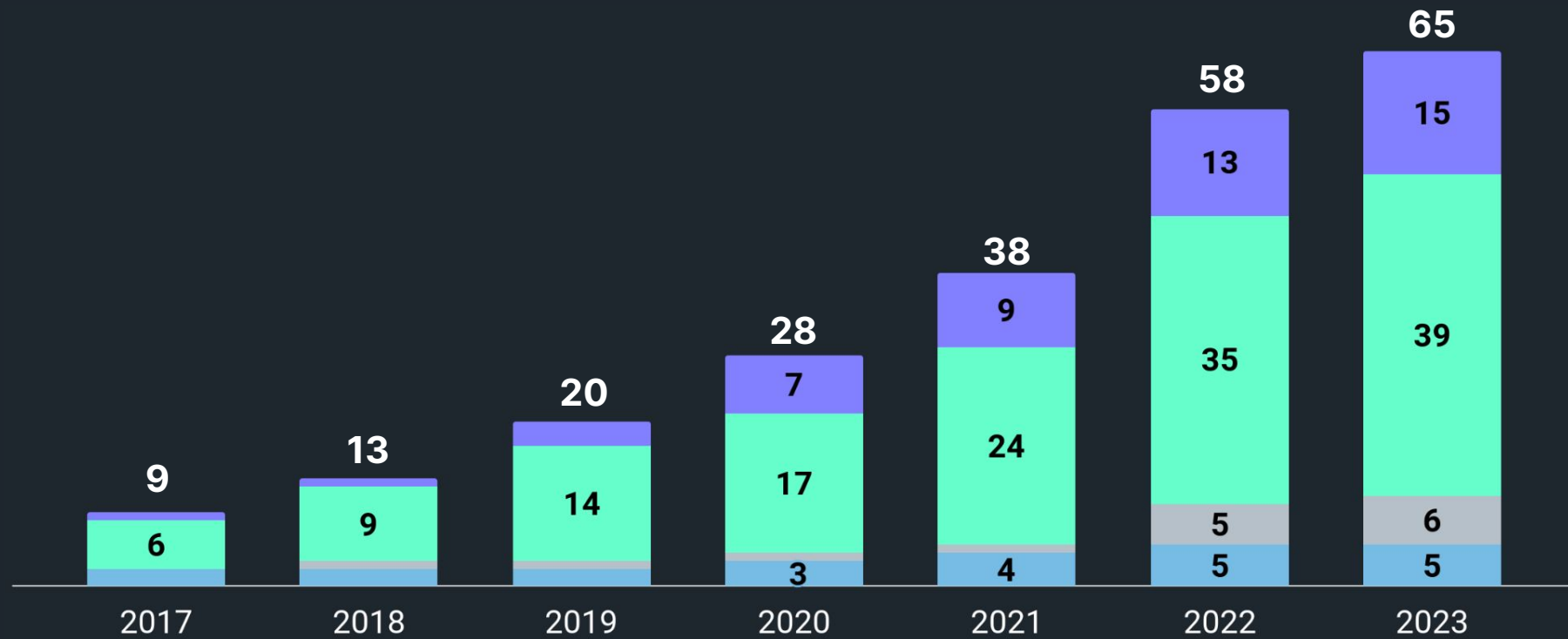
Head of Development  
Checkmk GmbH





- 01  The Checkmk tech organization
- 02  Increasing quality & security of Checkmk
- 03  Further product updates

# Our tech organization has grown a lot



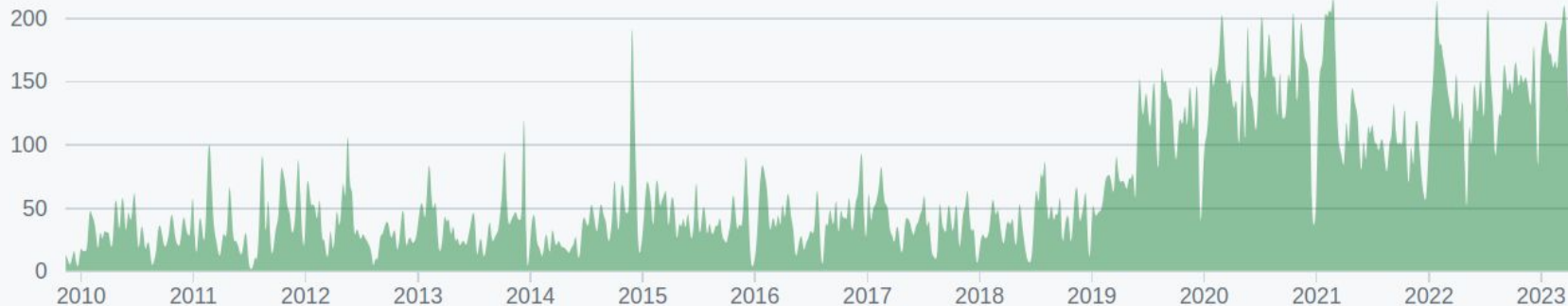
# Growth also visible in our Checkmk repository



Nov 15, 2009 – Jun 17, 2023

Contributions: Commits ▾

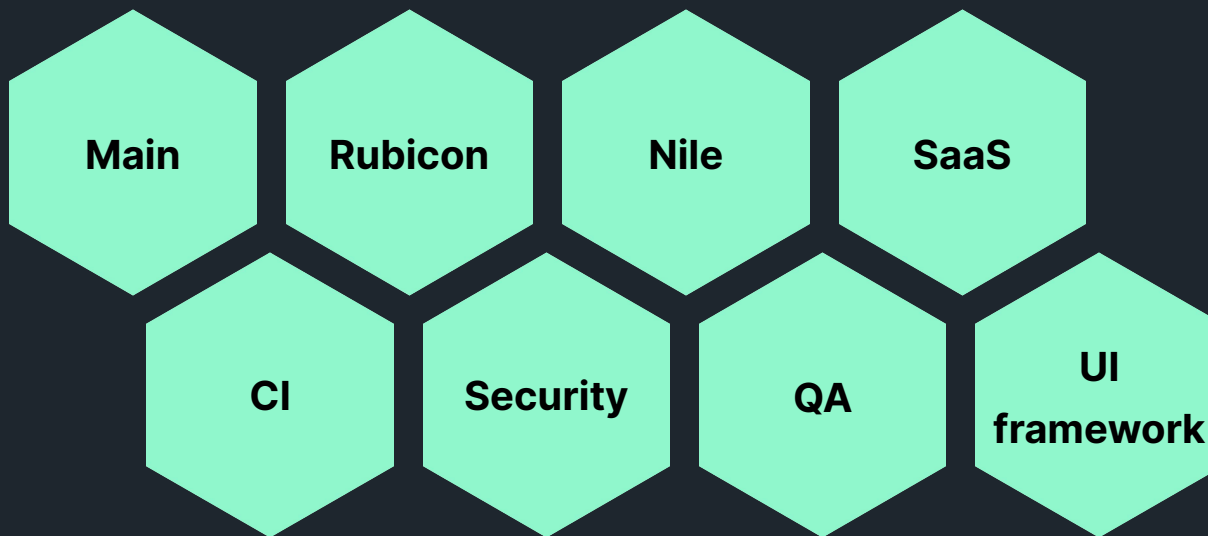
Contributions to master, excluding merge commits and bot accounts



# The Checkmk Engineering teams



Stream  
aligned  
teams



Platform and  
enabling  
teams

# Stream aligned teams: What they do

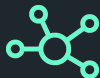


## Main

Improve core software elements, also Event Console, Bakery, etc.

### Example projects:

- Remove Python import cycles
- Self registration with bakery



## Rubicon

Not only UI, but also a lot of under-the-hood things.

### Example projects:

- REST-API
- Grafana connector
- SAML integration
- Label filter in views



## Nile

Monitoring the cloud, from the cloud.

### Example projects:

- TLS for agent
- Push mode for agent
- Auto. host registration



## SaaS

Build our SaaS platform

# Platform and enabling teams: What they do



## CI

Build Checkmk and developer tooling to increase efficiency of all teams



## Security

Product security and security-related processes.  
External and internal penetration tests.



## QA

Automated testing.  
Testing frameworks.  
Testing processes.



## UI framework

New UI framework for more performance, faster feature delivery & better maintainability

# The three pillars of Checkmk development



## Features



## Invest into the future

- Python2 → Python3
- Checker & helper split
- linting & typing
- separating backend from UI code
- splitting modules into packages



## Customer support

- Checks & agents improvements and bug fixes triggered via support tickets
- Component support (e.g. REST-API)
- Pull Requests



# Increasing quality & security of Checkmk

# Detect issues earlier



- Static code analysis
- Automated tests
- Testing by Checkmk consultants

# Improve quality via static code analysis ...



## Typing

### Python is a dynamically typed language

- ◆ Type errors are raised during runtime
- ◆ A lot of potential bugs remain hidden

### Using type hints and type checking

- ◆ Introduced typing in 2018, massively extended since then (2k+ commits)
- ◆ Helps catching a lot of bugs already during development

## Linting

### Flag programming errors, bugs, stylistic errors and suspicious constructs.

- ◆ Coverage and strictness increased strongly over the last years
- ◆ Reduces bugs, increases security and speed of development

# ... and unit and integration tests

# tests per version



# In-field testing for patch updates



For major release

**~15x**

during Alpha

Public beta

**Automated testing**

**Internal environments**

**Customer environments**

**Release**

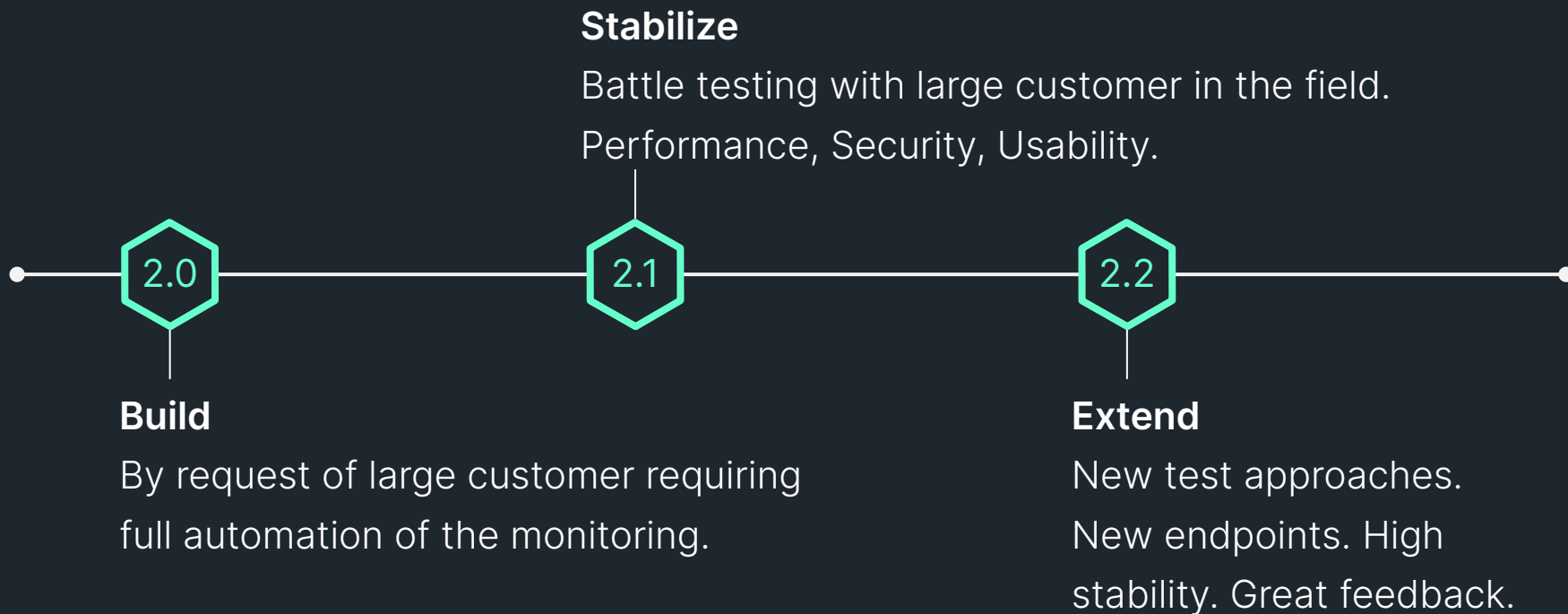
Automatic testing of release candidates (unit & integration tests)

Roll-out in our monitoring environment (~7k services)

Roll-out in 2-3 productive customer environments

Build and publish of release

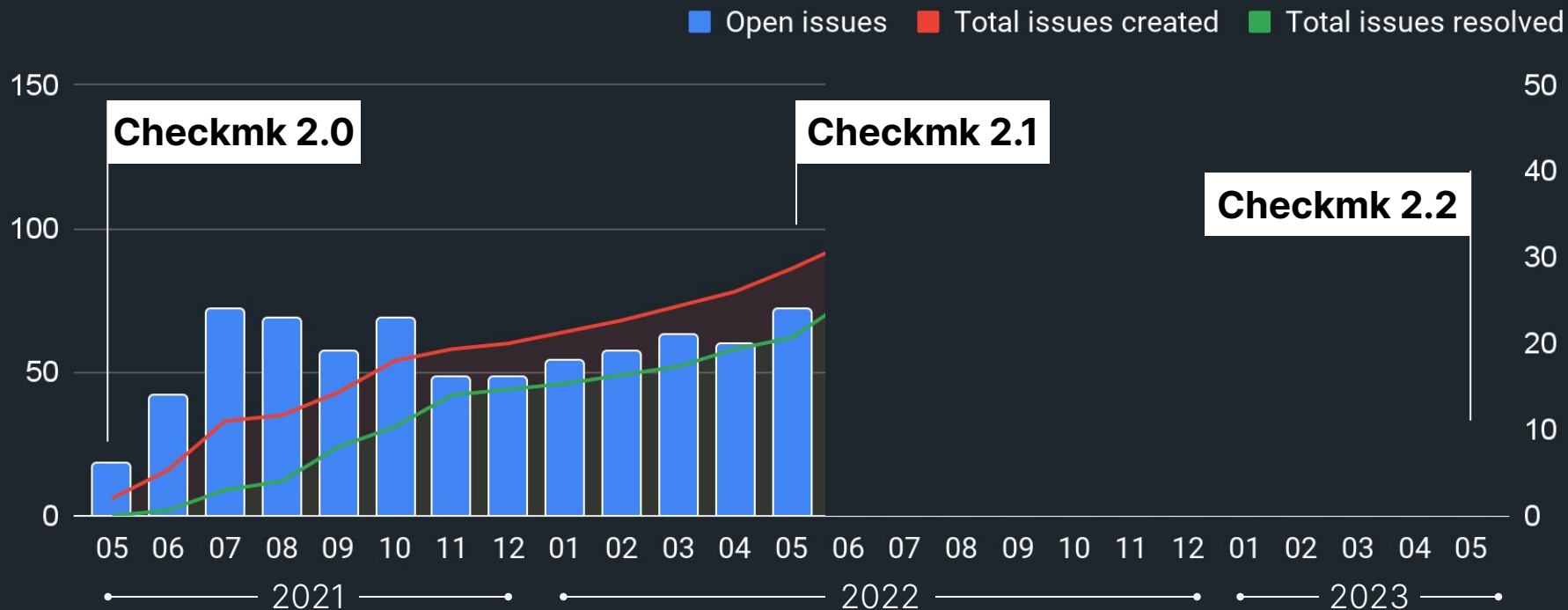
# The REST-API story



# The REST-API story: Only 15 new issues in 2023



## REST-API: 3rd Level



# New: Dedicated QA team



## Before

Tests are done by

- ◆ Developers: automated testing
- ◆ Consultants: Manual pre-release tests

## Now

New test team to support dev & consultant

- ◆ Dedicated test capacity
- ◆ Test tooling
- ◆ Test methods & experience
- ◆ 'Fresh external perspective'



# Current focus areas of the QA team



## Check plug-ins

---



## Checkmk updates combined with check plug-ins

Before vs. after update comparison (detect missing services etc.)

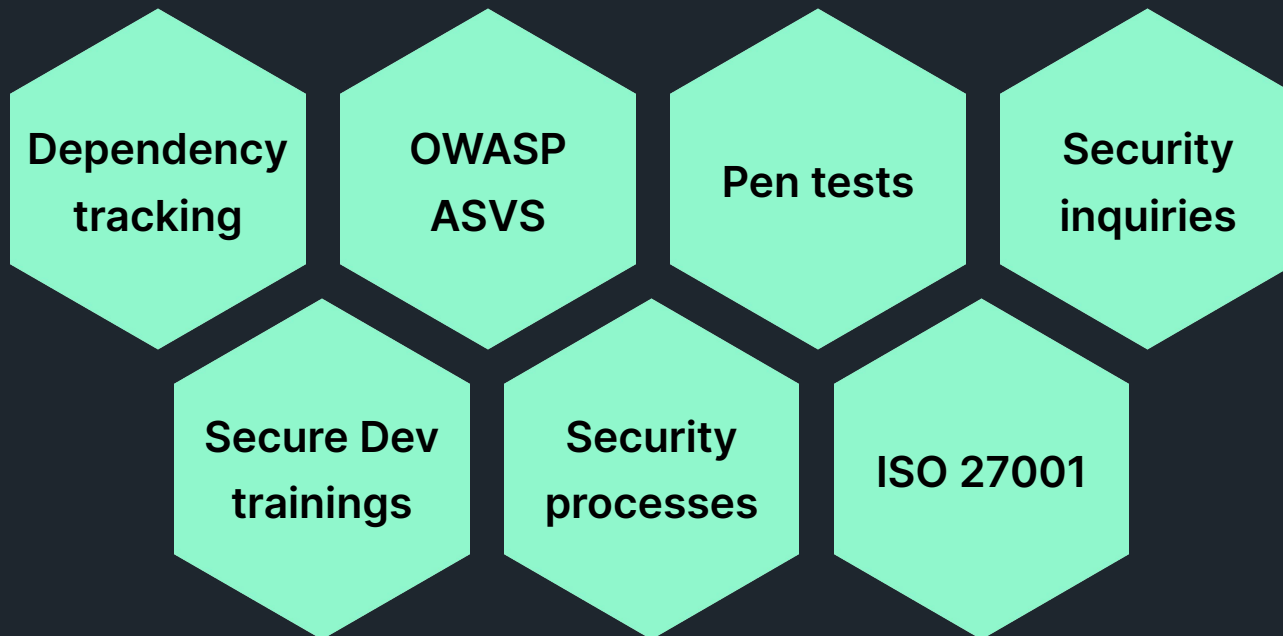
---



## MKPs

Loading of top 25 exchange plug-ins into Checkmk.

# Roadmap for Checkmk security



# Further product updates

# More reliability for supported environments



**Supported server OS**

---



**Supported browsers** and versions  
and mobile browsers

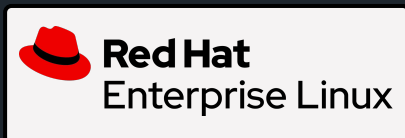
---



**Supported versions of integrations**, e.g. Grafana, Checkmk  
Appliance, LDAP server, SAML IdP, ntop, InfluxDB

[https://docs.checkmk.com/latest/en/release\\_notes.html](https://docs.checkmk.com/latest/en/release_notes.html)

# Four Linux distros supported for Checkmk Server



Maint. support 2



LTSS



ESM



LTS

# Checkmk server support guidelines



## The rules

- ◆ Support LTS versions only
- ◆ Support distro versions that are still in support 3 months after our initial major release
- ◆ Support distro versions which are initially released within our active maintenance
- ◆ Remove support for distro versions only for new major releases

## Exceptions are possible

- ◆ Not support a distro version, if we have strong technical reasons or there are more than 4 versions of a single distro in parallel in the field

# Checkmk support cycle for 2.2

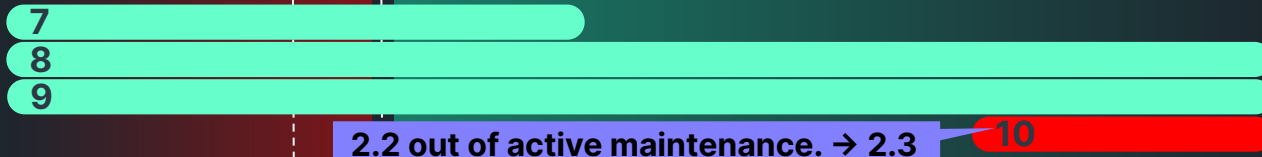


Checkmk 2.2 release date

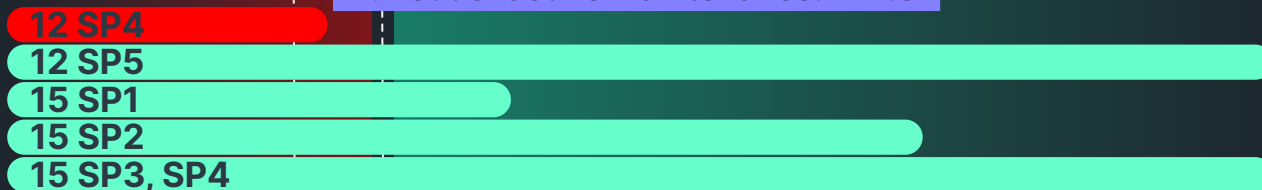
"Cut-off" date for disto version support



**Red Hat**  
Enterprise Linux



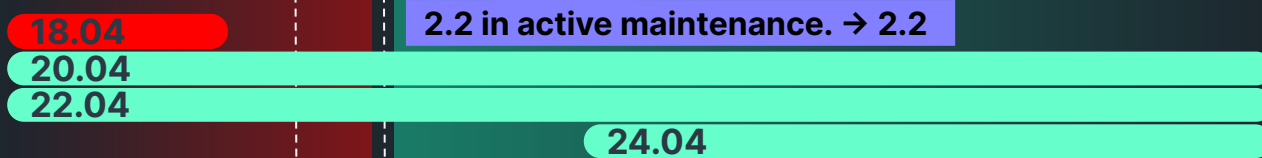
**suse**



**debian**



**Ubuntu**



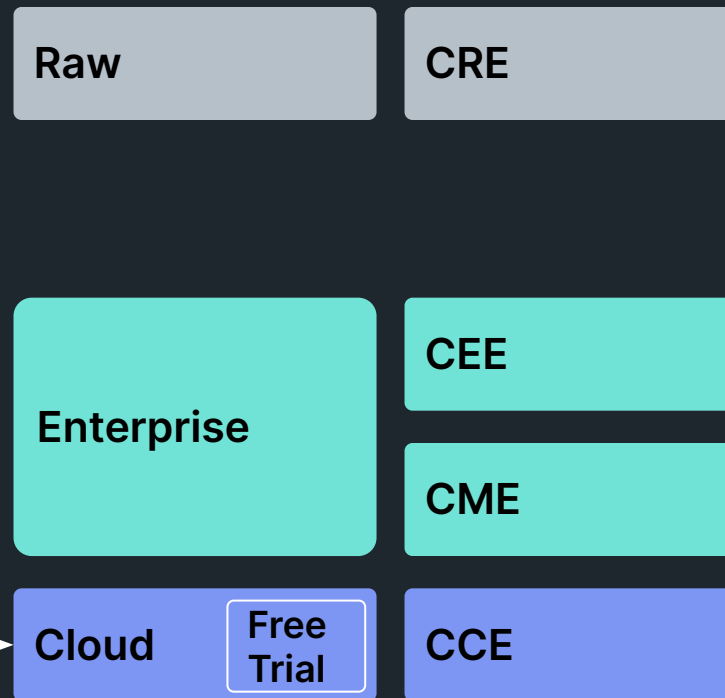
# Changes to Checkmk Editions



## Checkmk 2.1

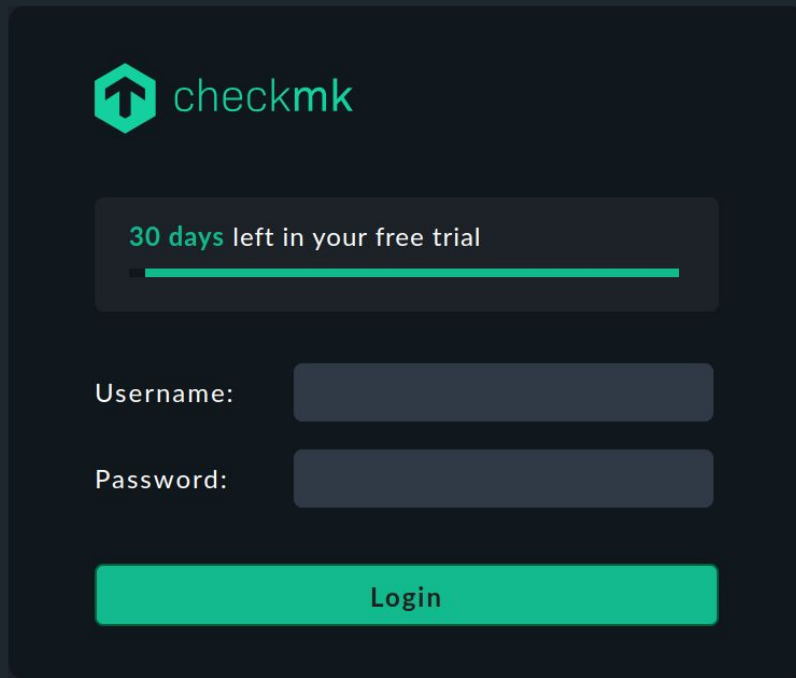


## Checkmk 2.2





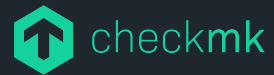
# The Checkmk Cloud Edition



- Checkmk Cloud includes a licensing feature
- Combination of a trial, free tier and full product now possible due to licensing feature
  - Easy transition to full product by entering license credentials
- Trial offering
  - 30 days (unchanged)
  - 1 site (unchanged)
  - 750 services (changed from 25 hosts)
    - Services are the Checkmk “currency”
    - 30 services per host is the standard
- Online and offline licensing supported

# Thank you!

Vielen Dank!



Checkmk GmbH  
Kellerstraße 27  
81667 München  
Germany

Web — [checkmk.com](https://checkmk.com)

E-Mail — [firstname.name@checkmk.com](mailto:firstname.name@checkmk.com)