

KEYCONF26 · PRAGUE · 8 OCTOBER 2026

The Keycloak Story

*From a JBoss side project to the CNCF's identity
standard*

Pritish Joshi

Technical Architect, Banfico

2014 → 26.8

WHERE THIS COMES FROM

At Banfico we use almost all of it.

Which is why I care how it got here.

Pritish Joshi
Technical Architect, Banfico

Open banking compliance

Seven countries

identity brokering · app-to-app · CIBA
CIBA since Keycloak 13.0

CoP / VoP

Confirmation and Verification of
Payee

DCR · DPoP · PKCE · OIDC
DPoP supported since 26.4

Trust Directory

OpenID Federation · other OAuth features

Banfico Identity Connect

EUDI wallet

OID4VCI · OID4VP
26.8: issuance preview, verifier
experimental

Every app rolled its own login

	BEFORE · EVERY APP	AFTER · KEYCLOAK, 2014
Login pages	One per application	One, shared by every app
Passwords	Stored and reset per app	Stored and reset once
Sessions	Each app keeps its own	Single sign-on across apps
OAuth, OIDC, SAML	Rebuilt, a little differently each time	Standard protocols, done once

The timing helped. APIs, mobile apps and SPAs made identity much harder, and open source meant anyone could download it, run it and contribute.

In 2014 you configured Keycloak. In 2026 you configure **OIDC**.

2014 · keycloak.json, read by the adapter

```
{  
  "realm": "demo",  
  "auth-server-url": "https://sso.acme.io/auth",  
  "ssl-required": "external",  
  "resource": "my-app",  
  "public-client": true  
}
```

2026 · any OIDC library, any language

```
issuer: https://sso.acme.io/realms/demo  
client_id: my-app  
response_type: code  
code_challenge_method: S256  
  
# the rest is discovered from  
# /.well-known/openid-configuration
```

The /auth prefix went in 17.0. Most Java adapters went in 25.0. What your app depends on is now a standard, not Keycloak.

From a server you configure to an image you build

- | | |
|------|---|
| 12.0 | Keycloak.X distribution, as a preview |
| 15.1 | Quarkus distribution, as a preview |
| 17.0 | Quarkus becomes the default. WildFly is deprecated and a 17.0.0-legacy image is published |
| 20.0 | WildFly distribution removed, three releases after it was deprecated |
| 22.0 | Quarkus 3 and Hibernate ORM 6. Java 11 dropped |

```
# Containerfile: the pattern from Keycloak's
container guide

FROM quay.io/keycloak/keycloak:26.8.0 AS builder
RUN /opt/keycloak/bin/kc.sh build

FROM quay.io/keycloak/keycloak:26.8.0
COPY --from=builder /opt/keycloak/ /opt/keycloak/
ENTRYPOINT ["/opt/keycloak/bin/kc.sh"]

# on every pod: build-time options are already baked
in

start --optimized
```

Build once in CI. Run the same image on every pod.

Where does session state live?

up to 24

In memory

Keycloak nodes

Infinispan: online sessions

DB: offline sessions only

Restart every node and online sessions are gone.

25 preview · 26.0 default

Persisted, and cached

Keycloak nodes

Infinispan: cache of sessions

DB: all user sessions

Sessions survive restarts. Each node caches 10,000 by default, the rest load from the DB.

--features=stateless

Never in memory

Site A

Site B

Caches: realm, client config

DB: all sessions, outbox

Sessions are database only, so sticky sessions stop mattering. Preview in 26.7, supported in 26.8.

multi-site: 23 preview → 24 supported → 26.0 active-active → 26.8 multi-cluster v2

Once the database holds the truth, indexes become the hot path, and 26.8 spends a lot of its fixes there.

Running it became part of the product

Upgrades	26.2 Rolling updates for optimized and customized images	26.3 Rolling updates for patch releases, as a preview	26.6 Zero-downtime patch releases, supported. Graceful HTTP shutdown	26.8 Missing indexes built in the background after an upgrade
Kubernetes	18.0 A rewritten Operator for the Quarkus distribution, as a preview	26.4 The Operator creates a ServiceMonitor for you	26.7 Kustomize install. Cluster-wide mode in preview	26.8 Client custom resources (preview). Helm chart (experimental)
Observability	21.0 Metrics move to Micrometer	25.0 Metrics and health on a separate management port	26.1 OpenTelemetry tracing supported	26.5 OpenTelemetry logs (preview), metrics (experimental). MDC logging supported

From rolling updates for optimized images in 26.2 to zero-downtime patch releases in 26.6: four minor releases.

Same SPIs, different everything else

```
// a provider factory, found by ServiceLoader
public class AuditListenerFactory
    implements EventListenerProviderFactory {
    public String getId() { return "audit"; }
    public EventListenerProvider create(
        KeycloakSession session) {
        return new AuditListener(session);
    }
}

# META-INF/services/org.keycloak.events.
#   EventListenerProviderFactory

$ cp audit.jar providers/
$ bin/kc.sh build
```

-
- 17.0 The jar goes in providers/ and kc.sh build bakes it in. SPI settings become CLI options

 - 22.0 Jakarta EE, Quarkus 3, Hibernate ORM 6, Java 11 dropped: a recompile

 - 23.0 The Map store is removed. The JPA store stays

 - 24.0 The User Profile SPI breaks. A new OAuth Grant Type SPI appears

 - 26.1 Provider factories can declare dependencies on each other

 - 26.8 Custom login themes need changes. The Password Policy SPI gets documented

What changed for people who build on it

-
- ~~The old admin console~~ gone in 21.0
 - + A React admin console on PatternFly since 19.0
-
- ~~Account Console v1 and v2~~ gone by 25.0
 - + Account Console v3, the default since 24.0
-
- ~~Java adapters, one per framework~~ most gone in 25.0
 - + Plain OIDC libraries. The Java admin client and keycloak-js get their own release cycles 26.0 and 26.1
-
- ~~Java EE packages, the Map store~~ gone in 22.0, 23.0
 - + Jakarta EE on Quarkus 3, and the JPA store 22.0, 23.0
-
- ~~An Arquillian and JUnit 4 test suite~~ replaced in 26.6
 - + The Keycloak Test Framework, on JUnit 6 supported 26.6
-
- The original Admin REST API for clients still there
 - + Client Admin API v2: strict validation, OpenAPI spec, Java and JS clients, a CLI preview 26.8
-

Keycloak didn't become important because JBoss created it. It became important because it **solved a universal problem**, embraced open standards, built a community, adapted to cloud-native architecture, and became bigger than its original home.

JBoss side project → Red Hat open source → developer adoption → cloud-native infrastructure → ecosystem → open governance

2014 · first release → 2016 · Red Hat SSO on Keycloak → 2022 · first outside maintainer → Apr 2023 · CNCF Incubating → Oct 2026 · 26.8

At every stage, Keycloak followed where application architecture was going next.