Honest Cheetah for GitHub — Privacy & Security

Last updated: February 2026

The Short Version

Honest Cheetah for GitHub stores only the **data needed to calculate flow metrics that GitHub doesn't natively expose** — primarily status change history with timestamps, calculated cycle times, and GitHub identifiers. Issue titles are retained for display and debugging. Current issue content (descriptions, comments, assignees) is always fetched directly from GitHub on demand. Your data is stored in Azure Cosmos DB with multi-tenant isolation and encrypted at rest and in transit.

How It Works

Honest Cheetah for GitHub is a **GitHub App** that you install at the organization level. Unlike the Azure DevOps version (which runs entirely within your AZDO tenant), the GitHub product has a hosted backend that processes and stores data:

- **GitHub App** — installed in your GitHub organization, providing scoped API access
- **Backend** — ASP.NET Core application hosted on Azure (App Service)
- **Database** — Azure Cosmos DB for persistent data storage
- **Authentication** — GitHub OAuth (users authenticate with their existing GitHub credentials)

What Data We Store

Data We Persist

We store only the data needed to calculate flow metrics that GitHub does not natively expose. Specifically:

Per Project Issue (ProjectIssue):

- GitHub identifiers (node IDs, issue number, project item ID, repository ID) — these are opaque IDs, not human-readable content
- Issue title (retained for debugging and display purposes)
- Status change history with timestamps (e.g., moved from "To Do" to "In Progress" at a specific date/time)
- Calculated flow metrics: cycle time, lead time, in-progress age, total age, throughput

completion date

- Issue metadata: labels, issue type, estimate values and estimate change history
- Issue state flags: archived, deleted, draft status

Per Project (Project):

- Project name, GitHub URL, organization login, and GitHub node IDs
- Project open/closed state and creation/modification dates
- Sync metadata (e.g., last backlog order sync timestamp)

Per Project Configuration (ProjectConfig):

- Metric definitions: which status values define your cycle time start/end, lead time start/end, and throughput completion
- Project field configurations: status values, priorities, sizes, issue types, and iterations
- Organization and project identifiers

What we don't store:

- Issue descriptions or body content
- Comments or discussion threads
- Assignee details
- Pull request or code data
- Repository contents
- User profile information beyond GitHub usernames for authentication

Other Stored Data

- **User account data** — GitHub user IDs and usernames for authentication and access control
- **GitHub App installation data** — installation IDs, organization mappings, and permissions granted

Webhook Data Lifecycle

GitHub sends webhook events to Honest Cheetah via the GitHub App when project items change. Here's how that data flows:

1. **Raw webhook event arrives** — the payload from GitHub is written to Cosmos DB as a `RawWebhookEvent` with a **1-hour time-to-live (TTL)**.
2. **Processing** — an Azure Function monitors the Cosmos DB change feed and extracts the relevant flow metrics data (status changes, timestamps, identifiers) from the raw event.
3. **Auto-deletion** — Cosmos DB automatically deletes the raw webhook payload after 1 hour. We do not retain the full GitHub webhook payload beyond this processing window.

4. **Processed data persists** — only the extracted flow metrics data (ProjectIssue records with status change history, calculated cycle times, etc.) is retained.

Separately, an optional webhook diagnostic log (`WebhookLogEntry`) exists for development debugging purposes. This logging is **disabled by default** in production and is only enabled in development environments.

Data We Fetch on Demand (Not Stored)

Current issue content — descriptions, comments, assignee details, and other live project data — is fetched directly from GitHub's API when needed and is not persisted in our database. The issue title is the only content field we retain.

Data We Encrypt

GitHub OAuth access tokens are encrypted before storage. These tokens are used to make API calls on behalf of authenticated users and are never stored in plaintext.

Multi-Tenant Data Isolation

All data in Cosmos DB is partitioned by GitHub organization ID (the organization's node ID serves as the partition key). This means:

- Each organization's data is logically isolated from every other organization's data
- Queries are scoped to the authenticated user's organization
- There is no mechanism for cross-tenant data access through the application

---

Authentication & Authorization

Honest Cheetah for GitHub uses a two-layer authentication model:

User Authentication

Users authenticate via **GitHub OAuth**. However you sign in to GitHub — including any MFA or SSO policies your GitHub organization enforces — that's what Honest Cheetah uses. We don't create separate accounts or passwords.

GitHub API Access

The application accesses GitHub data through two mechanisms:

- **GitHub App installation tokens** — for organization-level operations (webhooks, project data)
- **User OAuth tokens** — for operations that should be attributed to a specific user (write operations use the user's own token to maintain proper audit trails in GitHub)

Access Control

- Users are mapped to GitHub installations based on their organization membership
- The application verifies that the authenticated user has access to the requested GitHub installation before serving any data
- GitHub's own permission model is respected — if a user doesn't have access to a repository or project in GitHub, they won't see that data in Honest Cheetah

---

GitHub App Permissions

Honest Cheetah requests the minimum permissions necessary to read project data and track issue status changes. The full permission set is listed below — everything not listed is set to **No access**.

Repository Permissions (3 selected + 1 mandatory)

Permission	Access Level	Why We Need It
Metadata	Read-only *(mandatory)*	Required by GitHub for all Apps. Allows searching repositories and reading basic repository metadata.
Contents	Read-only	Read repository information needed to associate issues with repositories.
Issues	Read and write	Read issue data for flow metrics. Write access is reserved for planned future project management assistance features and is not currently used.
Projects	Read-only	Read classic project data at the repository level.

Organization Permissions (4 selected)

Permission	Access Level	Why We Need It
Issue Fields	Read-only	Read custom issue field definitions configured for the organization.
Issue Types	Read-only	Read issue type definitions (Bug, Task, Feature, etc.) configured for the organization.
Members	Read-only	Read organization membership to verify user access and associate users with installations.
Projects	Read and write	Read and manage organization-level GitHub Projects v2 data — the primary source of flow metrics. Write access enables updating project item fields.

Account Permissions

None. Honest Cheetah does not request any individual user account permissions.

Subscribed Webhook Events

The app subscribes to the following events to keep flow metrics data current:

Event	What It Tells Us
Installation target	GitHub App installation renamed — used to keep installation records in sync.
Issues	Issue opened, closed, reopened, labeled, transferred, and other state changes.
Projects v2	Project created, updated, deleted, closed, or reopened.
Projects v2 item	Project item created, edited, deleted, archived, restored, or reordered. This is the primary event for tracking status changes.
Projects v2 status update	Project status updates created, updated, or deleted.
Sub-issues	Sub-issues added or removed, and parent issue relationships changed.

Encryption & Credential Management

- **In transit:** All communications use HTTPS with TLS 1.2+. This includes browser-to-backend, backend-to-GitHub API, and backend-to-Cosmos DB connections.

- **At rest:** Azure Cosmos DB encrypts all data at rest using Microsoft-managed keys. GitHub OAuth access tokens receive an additional layer of application-level encryption before storage.
- **Service authentication:** The application authenticates to Azure Cosmos DB and Azure Application Insights using **Azure Managed System Identity** — no database credentials or connection strings are stored in application configuration or code.

Webhooks

Honest Cheetah receives webhook events from GitHub when project items change status. These webhooks are:

- Delivered over HTTPS
- Validated using GitHub's webhook signature verification (HMAC-SHA256)
- Processed as described in the **Webhook Data Lifecycle** section above — raw payloads are auto-deleted after 1 hour, and only extracted flow metrics data is retained

---

Hosting & Infrastructure

Component	Provider	Region	Compliance
Application hosting	Azure App Service	US East 2	Microsoft Azure compliance
Database	Azure Cosmos DB	US East 2	Microsoft Azure compliance
Telemetry	Azure Application Insights	US East 2	Microsoft Azure compliance
User authentication	GitHub OAuth	Per customer's GitHub plan	GitHub security
Source data	GitHub	Per customer's GitHub plan	GitHub security

Honest Cheetah builds on infrastructure provided by Microsoft Azure and GitHub. These providers maintain their own compliance certifications (including SOC 2, ISO 27001, and others). For current details, refer to the links above.

Data Residency

All Honest Cheetah infrastructure is hosted in the **Azure US East 2** datacenter region, including:

- Azure App Services (web applications and Azure Functions)
- Azure Cosmos DB (database)
- Azure Application Insights (telemetry)

GitHub data residency is determined by the customer's GitHub plan and configuration.

What We Don't Do

- ❌ We don't store issue descriptions, comments, or discussion content
- ❌ We don't access repositories, code, pull requests, or CI/CD pipelines
- ❌ We don't store passwords — authentication is handled entirely through GitHub OAuth
- ❌ We don't share data across organizations — multi-tenant isolation is enforced at the database level
- ❌ We don't sell or share your data with third parties
- ❌ We don't access financial, HR, or any non-project-management data

---

Auditing & Logging

- User authentication events (login, token validation) are logged server-side
- Raw GitHub webhook payloads are retained for 1 hour during processing, then automatically deleted by Cosmos DB TTL
- Webhook diagnostic logging is disabled by default in production and is only enabled for development debugging
- Application performance and error telemetry is collected via **Azure Application Insights** (Microsoft's application monitoring service) for diagnostics and reliability monitoring
- Application logs include user IDs and source context for troubleshooting
- Logs are stored in Azure-managed infrastructure with access restricted to authorized personnel

---

Data Deletion & Cancellation

When you uninstall the Honest Cheetah GitHub App from your organization:

- **Flow metrics data** (project issues, status change history, calculated metrics, and project configurations) will be deleted upon request. Contact support to initiate deletion.
- **User account data** associated with the organization will be deactivated and deleted within 90 days of cancellation.
- **Webhook data** — raw webhook payloads are auto-deleted after 1 hour regardless of account status. Webhook diagnostic logging is disabled by default in production.

- **GitHub OAuth tokens** are invalidated when the GitHub App is uninstalled and are deleted from our database.

You can request a full data export or deletion at any time by contacting support.

GDPR

For users subject to GDPR:

- We collect only the data necessary to operate the service — primarily GitHub node IDs, status change timestamps, and issue titles for calculating flow metrics, plus GitHub usernames for authentication
- You can request access to, correction of, or deletion of your personal data at any time
- Data processing details are available in our Data Processing Agreement (DPA), available upon request
- Our infrastructure providers (Microsoft Azure, GitHub) maintain their own GDPR compliance certifications

---

Questions?

Contact us at support@honestcheetah.com.