Benevity Teams App

Security & IT Admin Guide

Self-Help Reference for Enterprise Security and IT Administrators

June 2026 | Version 1.0

1. Introduction

This guide is written for the security architects, IT administrators, and compliance officers at organizations that use Benevity's corporate social responsibility (CSR) platform and are evaluating or deploying the Benevity Microsoft Teams App.

It is designed to answer the most common questions your security and IT teams will have — covering permissions, authentication, data handling, telemetry, infrastructure, and administrative controls — so you can make an informed deployment decision and configure the app to meet your organization's security requirements.

1.1 What is the Benevity Teams App?

The Benevity Teams App is a Microsoft Teams-native extension that allows employees to discover and participate in CSR activities — such as charitable giving, volunteer opportunities, and matching programs — directly within Microsoft Teams, without leaving their existing workflow.

Program administrators at your organization can publish opportunities to specific Teams channels. Employees can view, initiate, and complete CSR actions from within Teams on desktop, mobile, and web.

1.2 Deployment Model

Attribute	Details
Distribution	Microsoft Teams App Store (multi-tenant, organization-wide, or user-assigned)
App type	Microsoft Teams app with Personal Tab, Bot, and Messaging Extension capabilities
Bot hosting	Google Cloud Platform (Cloud Run) — serverless, auto-scaling container
Channel connector	Azure Bot Service — routes Teams messages to the bot endpoint
Identity provider	Microsoft Entra ID (Azure AD) — multi-tenant OAuth 2.0 / OIDC
Supported platforms	Teams Desktop (Windows/Mac), Teams Web, Teams Mobile (iOS/Android)
Target release	Beta: August 2026 | General Availability: September 2026

2. Microsoft Graph API Permissions

The Benevity Teams App requests the minimum set of Microsoft Graph API permissions necessary to deliver its functionality. This section provides a full mapping of every permission requested, its type, and its business justification.

2.1 Delegated Permissions (User Consent)

Delegated permissions operate on behalf of the signed-in user. They are granted at the time of first sign-in and scoped strictly to that user's own data.

Permission	Scope	Why We Need It
User.Read	Delegated	Read the signed-in user's profile (name, email, company, preferred language) to personalize the app experience..
openid	Delegated	Standard OpenID Connect scope required for authentication. Enables the app to verify the user's identity.
profile	Delegated	Read basic profile claims (display name, job title) included in the ID token at sign-in.
email	Delegated	Read the user's primary email address
offline_access	Delegated	Obtain a refresh token so the app can maintain the user's session without requiring re-authentication on every interaction.
Chat.ReadWrite	Delegated	Send and receive messages in personal and group chats where the bot has been added by the user.
ChannelMessage.Send	Delegated	Send messages to shared channels when a program admin publishes an opportunity. Write-only; the app does not read historical channel messages.
Team.ReadBasic.All	Delegated	Read the list of teams the signed-in user is a member of, so program admins can select which team to publish opportunities to. Read-only; no team settings are modified.
Channel.ReadBasic.All	Delegated	Read the list of channels in a team so program admins can select the target channel for publishing. Read-only.

2.2 Resource-Specific Consent (RSC)

For team and channel-scoped functionality, the app uses Resource-Specific Consent (RSC), which limits permission grants to specific teams or chats rather than the entire organization. The following RSC permissions may be requested when a program admin adds the app to a specific team:

RSC Permission	Scope	Purpose
TeamSettings.Read.Group	Team	Read team settings (name, description) for the selected team. Read-only.
ChannelMessage.Read.Group	Team	Read messages in channels where the app is explicitly installed. Only applicable if the admin installs the app into a specific channel.

3. Authentication & Identity

3.1 Sign-In Flow

The Benevity Teams App uses the industry-standard OAuth 2.0 Authorization Code flow with PKCE (Proof Key for Code Exchange) for user authentication. No passwords are transmitted to or stored by Benevity.

1. Employee opens the Benevity app in Teams.
2. The app redirects to Microsoft's identity platform (login.microsoftonline.com) for sign-in.
3. The user authenticates with their existing Microsoft 365 credentials — including any MFA policies your organization has configured.
4. Microsoft issues an access token and a refresh token scoped to the permissions listed in Section 2.
5. The app uses these tokens to call Microsoft Graph on behalf of the user. Tokens are never stored in the bot database — they are cached in memory for the duration of the session only.

3.2 Multi-Tenant Architecture

The app is registered as a multi-tenant application in Microsoft Entra ID. This means:

- Each customer organization has its own isolated sign-in flow via its own Entra ID tenant.
- User data from one tenant is never accessible to users or admins in another tenant.
- Your organization's Conditional Access policies, MFA requirements, and device compliance policies are enforced by Microsoft at sign-in — Benevity does not bypass these controls.

> **Single Sign-On:** *The app supports Microsoft Teams SSO for tab experiences. Users who are already signed into Teams are not prompted to sign in again — Teams silently passes an authentication token to the app, providing a seamless experience without additional credential entry.*

3.3 Token Handling

Token Type	Lifetime	Storage
Access token	1 hour (Microsoft default)	In-memory cache on Cloud Run — never written to disk or database
Refresh token	Up to 90 days (sliding)	Encrypted, stored in GCP Secret Manager — per-user, per-tenant isolated

Token Type	Lifetime	Storage
ID token	Single-use at sign-in	Not persisted after initial identity verification

3.4 Your Conditional Access Policies Are Respected

Because authentication is handled entirely by Microsoft Entra ID, all Conditional Access policies you have configured in your tenant apply automatically to the Benevity Teams App. This includes:

- Multi-factor authentication (MFA) requirements
- Device compliance and Intune-managed device policies
- Location-based access restrictions (named locations, IP ranges)
- Sign-in risk policies (Microsoft Entra ID Protection)

Benevity does not implement a separate identity layer that could circumvent these controls.

4. Data Handling & Privacy

4.1 Data Collected by the App

Data Category	Examples	Purpose	Retention
User profile	Name, work email, company, preferred language	Personalize app experience; pre-fill forms	Session only; not stored persistently
CSR transaction data	Donation amount, charity selected, volunteer event registered	Process and confirm CSR actions; generate tax receipts	Retained in the Benevity platform per your data processing agreement
Teams context	Team name, channel name, chat roster	Route published opportunities to the correct channel	Not persisted – used transiently during publish action
Telemetry events	Anonymized event IDs (e.g., opportunity viewed, action initiated)	Product analytics and system health monitoring	90 days in operational logs; up to 2 years in analytics data lake

4.2 Data NOT Collected

> **Benevity does not collect:** *Email message content, file contents, calendar events, personal chat messages outside of bot interactions, browsing history, device information, or any data outside the explicit permissions listed in Section 2.*

4.3 Data Residency & Storage

The Benevity Teams App infrastructure runs on Google Cloud Platform (GCP). Data residency regions are configured per your organizational agreement with Benevity. By default:

Data Type	Location	Provider
Bot runtime (in-memory)	GCP us-central1 (or contractual region)	Google Cloud Run
Application secrets	Azure Key Vault (same region as your Azure agreement)	Microsoft Azure
Telemetry/analytics events	GCP Cloud Storage (Data Lake) + BigQuery	Google Cloud Platform
CSR transaction records	Benevity platform (region per your DPA)	Benevity (AWS or GCP)

5. Telemetry & Usage Analytics

The Benevity Teams App collects product telemetry to measure feature adoption, identify usability issues, and monitor system health. This section describes exactly what is collected, how it is anonymized, and where it goes.

5.1 Telemetry Events Collected

Event	What is Captured	What is NOT Captured
Opportunity Viewed	Event type, timestamp, surface (channel/chatbot/tab), anonymized opportunity ID	Employee name, personal details, browsing context outside Teams
Action Initiated	Event type, timestamp, action type (donate/volunteer)	Amount, charity, or any transaction content
Inputs Submitted	Event type, timestamp, form completion indicator	Form field values, amounts, and personal inputs
Validation Failed	Error category (not error message content), timestamp	User input content
Transaction Submitted	Event type, timestamp, transaction reference (opaque ID)	Amount, payment method, and personal data
Outcome Resolved	Outcome category (success/failed/blocked), timestamp, duration	Transaction content or personal data
Redirection to Web	Event type, timestamp	Destination URL parameters or user data
Admin Publish Action	Event type, timestamp, success/failure flag, tenant ID	Channel content or employee data
Interaction Latency	Response time in milliseconds	No user data – performance metric only

5.2 Anonymization

- User IDs in telemetry are hashed (SHA-256 with a per-tenant salt) before being written to the analytics pipeline. Benevity cannot reverse these hashes to identify individual employees.
- Tenant IDs are included to enable per-organization analytics, but are stored separately from event payloads in the data lake.
- No message content, form values, amounts, or personal data is included in any telemetry event.

5.3 Telemetry Pipeline

Telemetry events flow from the Cloud Run bot service through the following pipeline:

6. Events emitted via OpenTelemetry SDK from the bot service.
7. Streamed to GCP Cloud Pub/Sub (topic: teams-app-telemetry).

8. Processed and validated by Cloud Dataflow (schema validation, deduplication, Parquet serialization).
9. Stored in GCP Cloud Storage (Data Lake) partitioned by date, tenant, and event type.
10. Queryable via BigQuery for product analytics and health dashboards.

Operational metrics (error rates, latency) are sent to GCP Cloud Monitoring with alert policies; no user data is included in these metrics.

6. Infrastructure & Security Controls

6.1 Architecture Overview

Component	Technology	Security Control
Bot service runtime	GCP Cloud Run (serverless containers)	No persistent state; ephemeral containers; auto-scaled; HTTPS enforced (TLS 1.2+)
Teams channel connector	Azure Bot Service (channel registration only – no code hosted here)	HMAC signature verification on every incoming Teams message
Secrets management	Azure Key Vault (RBAC-controlled)	No secrets in code or environment variables; Managed Identity access; audit logs retained 90 days
Container images	GCP Artifact Registry	Vulnerability scanning enabled; zero critical CVEs policy before deployment
Bot ↔ Key Vault auth	GCP Workload Identity Federation	No long-lived Azure passwords stored in GCP; keyless cross-cloud authentication
CI/CD pipeline	GitHub Actions + GCP Workload Identity	No long-lived service account keys; OIDC-based authentication; production requires 2-reviewer approval

6.2 Network Security

- All communication between the Teams client and the bot service uses HTTPS (TLS 1.2 minimum, TLS 1.3 preferred).
- The bot endpoint on Cloud Run validates the Bearer token on every incoming request from Azure Bot Service before processing any message.
- HMAC signature verification is applied to all Bot Framework activity payloads to prevent spoofing.
- CORS is configured to allow only known Microsoft Teams origins.
- The Cloud Run service does not expose any administrative APIs publicly.

6.3 Vulnerability Management

- All container images are scanned for known CVEs via GCP Artifact Registry scanning before deployment.
- Dependency vulnerability scanning (npm audit) runs in every CI/CD pipeline execution. Critical vulnerabilities block deployment.
- Benevity's security team conducts quarterly penetration tests on the Teams App infrastructure.
- Security patches are applied within 30 days of disclosure for medium/high severity; within 7 days for critical.

6.4 Compliance & Certifications

Benevity maintains the following certifications relevant to the Teams App. Current certificates are available upon request from your Customer Success Manager.

Certification / Standard	Status	Scope
SOC 2 Type II	Certified	Benevity platform and infrastructure, including Teams App services
GDPR	Compliant	Data processing practices; DPA available for execution
Microsoft Teams App Certification	Pending (GA target)	Microsoft Publisher Attestation Program
ISO 27001	In scope for the CY 2026 audit	Information security management system

7. IT Admin Controls & Deployment

7.1 Installing the App in Your Tenant

As a Teams Administrator, you control whether and how the Benevity Teams App is available to users in your organization.

11. Sign in to the Microsoft Teams Admin Center (admin.teams.microsoft.com) with Teams Administrator credentials.
12. Navigate to Teams apps → Manage apps and search for "Benevity".
13. Review the app permissions and publisher attestation before allowing.
14. Set the app status to Allowed for your organization.
15. Navigate to Teams apps → Permission policies to control which users can install the app.
16. Navigate to Teams apps → Setup policies to optionally pin the app for specific user groups.

7.2 Restricting the App to Specific Users

To limit the app to a pilot group or specific departments before an organization-wide rollout:

17. In Teams Admin Center → Teams apps → Permission policies, create a new policy.
18. Under Microsoft apps and Third-party apps, set to "Allow specific apps" and add Benevity.
19. Assign the policy to the target user group via Teams admin or Entra ID group assignment.
20. Users outside the assigned group will not see the app in the Teams store.

7.3 Removing the App

To remove the Benevity Teams App from your organization:

21. In Teams Admin Center → Teams apps → Manage apps, set the app status to Blocked.
22. Existing bot conversations and channel messages will remain, but the app will no longer be functional.
23. To remove the app's consent grants, navigate to Microsoft Entra ID admin center → Enterprise applications → search for Benevity → Permissions → Revoke admin consent.

Revoking consent will sign all users out of the app immediately. User consent grants will also be cleared, requiring users to re-consent on their next sign-in if the app is re-enabled.

7.4 Data Deletion & Off-boarding

When an employee leaves your organization, their Microsoft 365 account is disabled or deleted in Entra ID. The Benevity Teams App will automatically reject any token refresh attempts for that user, effectively revoking their app access within one hour (the access token lifetime).

To request deletion of a specific user's personal data held by Benevity in accordance with GDPR Article 17 (right to erasure), submit a data subject access request via your Benevity Customer Success Manager or at privacy@benevity.com.

8. Frequently Asked Questions

Can the app read our employees' emails or Teams messages?

No. The app does not request Mail.Read or any permission to access email content. It can only send messages to chats and channels where it has been explicitly added, and it can read basic roster information (names and emails of chat members) within those specific chats. It cannot read historical messages.

Does the app store Microsoft 365 credentials or passwords?

No. The app uses OAuth 2.0 and never receives or stores passwords. Authentication is handled entirely by Microsoft's identity platform. The app only receives access tokens and refresh tokens, which are time-limited and revocable.

Can Benevity employees access our users' data?

Benevity engineering and support staff access production systems only via break-glass procedures with full audit logging. Access is role-based, requires approval, and is logged to an immutable audit trail. Benevity does not perform ad-hoc queries on customer data outside of agreed support scenarios.

Is the app available in Government or GCC High environments?

The current release supports Microsoft 365 commercial cloud tenants. Government Community Cloud (GCC) and GCC High support is on the roadmap. Contact your Benevity account team for current availability and timelines.

Does the app work with our MFA / Conditional Access policies?

Yes. Because authentication is handled by Microsoft Entra ID, all your existing Conditional Access policies — including MFA, device compliance, and location-based restrictions — apply automatically. Benevity has no ability to bypass these controls.

Where is the telemetry data stored, and can we request it be excluded?

Telemetry is stored in GCP Cloud Storage and BigQuery in the region specified in your agreement (default: US). All telemetry events are anonymized before storage — employee-identifiable data is hashed. Enterprise customers can discuss telemetry reduction options with their Customer Success Manager. See Section 5 for full details.

What happens if Benevity has a security incident?

Benevity will notify affected customers within 72 hours of confirming a personal data breach, in accordance with GDPR requirements and your Data Processing Agreement. A dedicated incident response team and runbook are maintained. Notification will be sent to the security contact registered in your Benevity account.

Does the app support DLP (Data Loss Prevention) policies?

The app communicates through standard Microsoft Teams bot framework channels, which are subject to your organization's Teams DLP policies configured in the Microsoft Purview compliance portal. Benevity does not intercept or bypass DLP controls.

9. Contacts & Further Information

Query Type	Contact	Response SLA
Security questions & vulnerability reports	security@benevity.com	1 business day
Data privacy & GDPR / data subject requests	privacy@benevity.com	3 business days
Penetration test approval requests	security@benevity.com	5 business days
Data Processing Agreement (DPA) requests	Your Benevity Customer Success Manager	3 business days
General IT admin support	support.benevity.com	Per your support tier SLA
Compliance certificates (SOC 2, ISO)	Your Benevity Customer Success Manager	5 business days

9.1 Useful Links

- Microsoft Teams Admin Center: https://admin.teams.microsoft.com
- Microsoft Entra ID Admin Center: https://entra.microsoft.com
- Benevity Trust & Security: https://benevity.com/security
- Benevity Privacy Policy: https://benevity.com/privacy
- Microsoft Graph Permissions Reference: https://learn.microsoft.com/en-us/graph/permissions-reference