

Paylocity Termination Backend (v1)

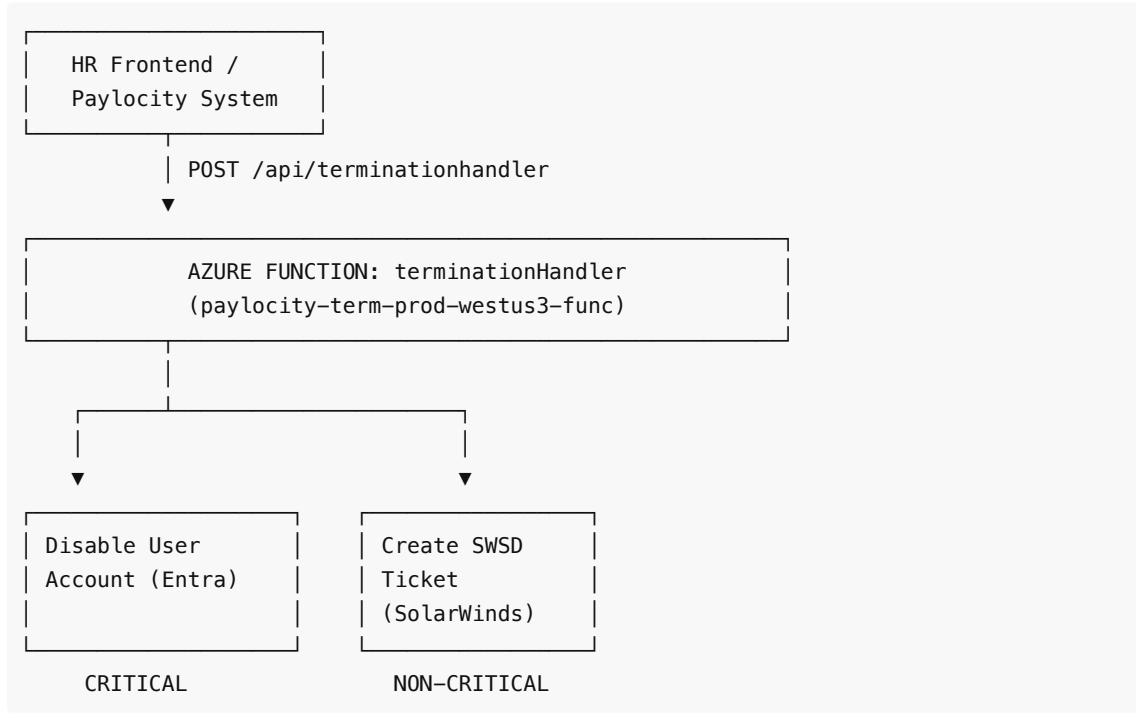
Executive Summary

The Paylocity Termination Backend is an **Azure Functions-based webhook service** that automates employee offboarding by disabling user accounts in Microsoft Entra ID when terminations are processed. It includes safety mechanisms to protect critical accounts and creates audit trail tickets in SolarWinds Service Desk for IT follow-up on additional offboarding tasks.

Business Value

Benefit	Description
Immediate Access Revocation	Terminated employees lose access within seconds
Security Compliance	Prevents unauthorized access post-termination
Audit Trail	Service desk tickets document all terminations
Protected Accounts	Deny list prevents accidental disabling of critical accounts
Multi-Tenant	Supports sandbox testing and production environments

Architecture Overview



Workflow Steps

Step 1: Validate Request

- Check for required fields: `companyId`, `employeeFirstName`, `employeeLastName`
- Return 400 if validation fails

Step 2: Environment Selection

- `companyId = "CATALYST"` → Sandbox environment
- Other company IDs → Production environment

Step 3: Search for User (Primary Method)

- Search Entra ID by `displayName` using `$search` parameter
- Query: `"displayName:FirstName LastName"`
- Requires `ConsistencyLevel: eventual` header

Step 4: Search for User (Fallback Method)

- If primary search fails, use `$filter` parameter
- Query: `givenName eq 'FirstName' and surname eq 'LastName'`
- Handles naming inconsistencies

Step 5: Deny List Check

- Compare user email against `TERMINATION_DENY_LIST`
- Return 403 if user is protected
- Prevents accidental disabling of critical accounts

Step 6: Disable User Account

- PATCH request to set `accountEnabled: false`
- User immediately loses access to all Microsoft services

Step 7: Create Service Desk Ticket

- Document termination in SolarWinds Service Desk
- Create audit trail for compliance
- Enable IT follow-up on additional offboarding tasks

API Contract

Webhook Endpoint

POST `https://paylocity-term-prod-westus3-func.azurewebsites.net/api/terminationhandler`

Request Body

```
{  
  "companyId": "163160",  
  "employeeFirstName": "John",  
  "employeeLastName": "Doe",  
  "employeeWorkEMailAddress": "john.doe@catalystsolutions.com",  
  "employeeCostCenter1": "Engineering",  
  "employeeJobTitle": "Software Developer",  
  "employeeTerminationDate": "2026-01-20T00:00:00Z"  
}
```

Response Codes

Status	Description
200	Success - user account disabled
400	Bad Request - missing required fields
403	Forbidden - user is on deny list (protected)
404	Not Found - user not found in Entra ID
500	Server Error - processing failure

Success Response

```
{  
  "success": true,  
  "disabledUser": "user-uuid",  
  "displayName": "John Doe"  
}
```

Protected User Response (403)

```
{  
  "success": false,  
  "message": "This user is protected and cannot be terminated via webhook"  
}
```

External Service Integrations

Microsoft Entra ID (Azure AD)

- **Authentication:** OAuth 2.0 Client Credentials via MSAL
- **API:** Microsoft Graph API
- **Operations:** User search, account disabling

SolarWinds Service Desk

- **Authentication:** Bearer token (X-Samanage-Authorization)
- **Base URL:** <https://catalystsolutions.samanage.com>
- **Operations:** Incident ticket creation

Azure Application Insights

- **Purpose:** Monitoring and logging
- **Events:** Function execution, errors, termination events

Environment Configuration

Environment	Company ID	Azure Tenant	Domain

Sandbox	CATALYST	ab558d63-c75b-4100-ae33-9160bbcfbaa	healthplansai.dog
Production	163160	d39a588a-b5a6-4378-9f5f-f9a0e5484b06	catalystsolutions.com

Security Features

Deny List Protection

- **Environment Variable:** TERMINATION_DENY_LIST
- **Format:** Comma-separated email addresses
- **Example:** admin@company.com, servicedesk@company.com
- **Behavior:** Returns 403 Forbidden if terminated user is on list

Why Deny List Matters

- Prevents accidental disabling of service accounts
- Protects admin accounts from webhook-based termination
- Adds safety layer for critical system accounts

Technical Stack

Component	Technology
Runtime	Azure Functions (Node.js 22.x)
Authentication	@azure/msal-node
HTTP Client	Axios
Monitoring	Application Insights
Deployment	GitHub Actions
Region	Azure West US 3

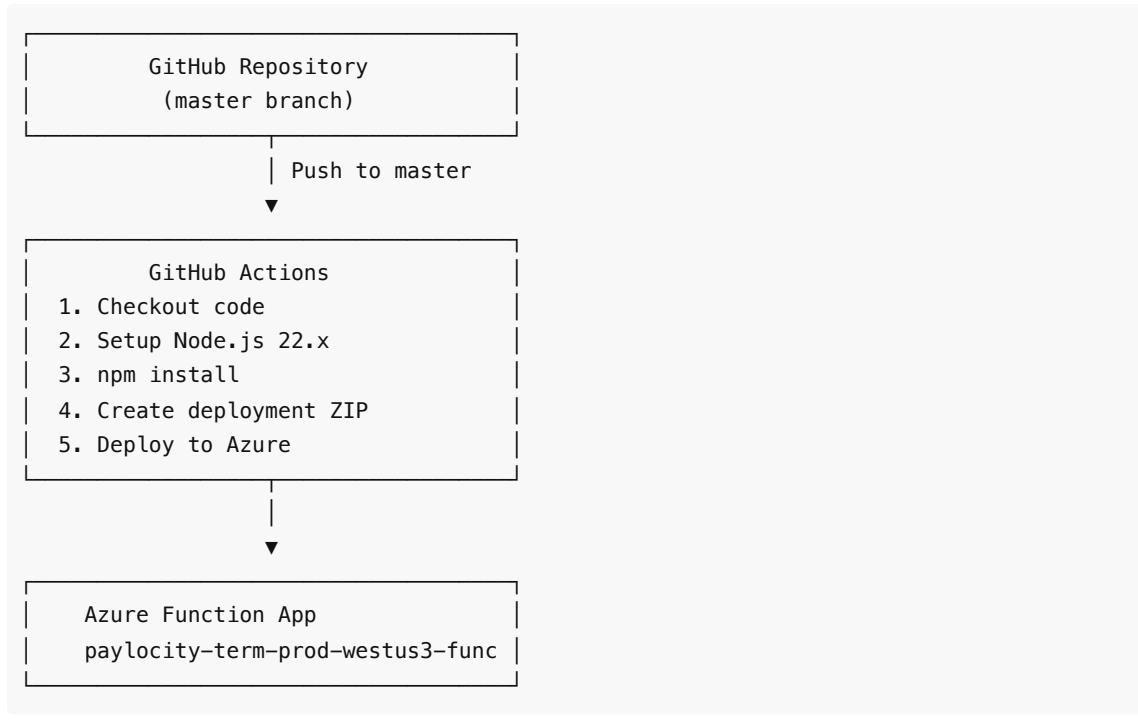
Project Structure

```

termination-v1/
├── TerminationWebhook/
│   ├── index.js                  # Main function handler
│   └── function.json             # Azure Function binding
├── shared/
│   ├── graphClient.js           # Entra ID/Graph API client
│   ├── swsdClient.js            # SolarWinds client
│   ├── env.js                   # Environment configuration
│   └── logger.js                # Application Insights logging
├── host.json                  # Azure Functions config
└── package.json                # Dependencies
└── .github/workflows/
    └── master_paylocity-term-*.yml # CI/CD

```

Deployment Pipeline



User Search Strategy

The system uses a dual-method approach to find users:

Primary Search (displayName)

```
GET /users?$search="displayName:John Doe"
Header: ConsistencyLevel: eventual
```

Fallback Search (givenName + surname)

```
GET /users?$filter=givenName eq 'John' and surname eq 'Doe'
```

This approach handles:

- Users with different display name formats
- Partial name matches
- Naming inconsistencies between systems

Service Desk Ticket Details

Ticket Properties:

- **Name:** TERMINATION - [FirstName] [LastName] - Blocked Entra Sign In
- **Priority:** 2 (Medium)
- **Category:** Account Management

- **Subcategory:** Terminations
- **State:** New
- **Requester:** servicedesk@catalystsolutions.com

Description Includes:

- Employee name and email
- Job title and department
- Termination date
- Instructions for additional offboarding tasks

What Gets Disabled

When the termination webhook fires:

Service	Access Revoked
Microsoft 365	Yes
Azure Portal	Yes
Microsoft Teams	Yes
Outlook/Exchange	Yes
SharePoint/OneDrive	Yes
Any Azure AD SSO app	Yes

The account is disabled, not deleted. Administrators can:

- Re-enable if termination was in error
- Access mailbox for legal holds
- Retrieve OneDrive files

Logging & Monitoring

Console Logging

- Emoji-based detailed logging for debugging
- Logs all major workflow steps
- Error details captured

Application Insights

- Function execution telemetry
- Error tracking
- Custom events for termination processing

Summary

The Termination Backend provides a critical security function: immediately revoking access when employees are terminated. Its dual-search strategy handles naming inconsistencies, while the deny list protects critical

accounts from accidental disabling. The integration with SolarWinds Service Desk ensures a complete audit trail and enables IT teams to handle additional offboarding tasks.

Document generated: January 2026