Table of Contents

⚠️ Important: Testing and Deployment Notice

Before enabling any Transaction Security Policy (TSP) provided in the TSP Accelerator managed package within a production environment, it is critical to conduct comprehensive testing in a non-production environment, such as a Sandbox or Scratch Org. Because these policies evaluate events in real-time–including logins, API queries, and report exports –improperly configured logic or missing metadata allowlists can result in immediate and unintended service disruptions or "lock-out" scenarios for your users. Always validate that policy triggers and configuration dependencies align with your specific organizational workflows and security requirements before final activation.

📋 Prerequisites

- Salesforce Event Monitoring or Shield
    - *The package installation will fail if you do not have Salesforce Event Monitoring or Shield licensed in the org*
- Hyperforce (as the app depends on Event Log Objects)
- System Administrator or equivalent permissions
- Lightning Experience

🔧 Setup

1. Package Installation
Install package from AppExchange

2. Configure Authentication
See the detailed Authentication Setup Guide below for step-by-step instructions on configuring:

- External Client App
- Auth Provider
- Named Credential

3. Assign permissions to users who will use the app

Because this is a managed package, certain system/user permissions cannot be included in the packaged permission set and must be granted separately in your org. Follow the steps below based on the user type.

For System Administrators

System Administrators will generally already have the permissions needed to run the app. However, they must be granted one additional permission that is new with this package: Modify Transaction Security Policies.

For non-Administrator users

1. Assign the included permission set:
    1. Navigate to Setup → Permission Sets
    2. Find and open the TSP Accelerator permission set
    3. Click Manage Assignments and assign it to the appropriate users

       This grants the packageable access: Apex class access, custom object and field permissions (e.g., RecordAccessLog__c), custom metadata type access, and the app tab.
2. Grant the system permissions that cannot be packaged.
    1. These are stripped out of managed-package permission sets by Salesforce, so you must enable them manually – either by creating a local permission set in your org and assigning it, or by enabling them on the users' profile. Enable the following under System Permissions:

- API Enabled
- Modify Transaction Security Policies
- Customize Application
- View All Data
- View Event Log Files
- View Platform Events
- View Setup and Configuration
- View Real-Time Event Monitoring Data

📖 Usage

Browsing Policies

1. Navigate to the TSP Accelerator tab
    - ***Note*** **-** The TSP Accelerator can be added to the App of your choice via App Manager in Setup
2. Policies are organized by security category:
    - **Data Protection & Classification**: Preventing exposure of sensitive or regulated data, such as PII or financial records.
    - **Access Control (Authentication)**: Monitoring login behavior and identifying suspicious or unauthorized access attempts.
    - **Security Model (Authorization)**: Ensuring that user permissions, profiles, and role-based access are used appropriately.
    - **Integration**: Governing data access and behavior of third-party apps and API-based systems.
    - **Data Loss Prevention (DLP):** Detecting attempts to extract data in bulk or bypass organizational controls.
    - **Monitoring**: Enabling visibility into security-related activity and surfacing meaningful alerts for investigation.

Deploying a Policy

1. Click the **Deploy** button on any policy card
2. Review the policy details in the modal:
    - Description
    - Why This Matters
    - Triggering Event
    - Configuration Requirements
3. Click **Configure & Deploy**
4. Edit the Policy Name and Description if needed
5. Click **Deploy**
6. The policy will be deployed as **Inactive** (you can activate it later in Setup)

Viewing Deployed Policies

- Click **View in Setup** at the bottom of the policy card
- The enabled / not enabled state of a policy is also shown in the policy card
- The Deploy button is disabled for already-deployed policies
    - Click the information button to get the policy details about an already-deployed policy
- To view all TSPs navigate to **Setup → Transaction Security Policies**

🔐 Security Considerations

- All policies are deployed as **Inactive** by default
- Review and test each policy before activating in production
- Policies require corresponding Apex classes or Flows to function

- Some policies require additional setup (see Configuration field in modal)

⚠️ Troubleshooting

Installation Error
If you receive installation error messages related to “ApexClass Invalid type”, this is because Salesforce Event Monitoring or Shield are not licensed in the org.
Salesforce Event Monitoring or Shield must be licensed in the org in order to successfully install the app.

Authentication and Deployment Errors
For authentication-related errors (Unauthorized endpoint, Invalid callback URL, Authentication failed), refer to the Authentication Setup Guide for detailed troubleshooting steps.

Policy Deploys but Doesn't Work
If a policy deploys successfully but doesn't function as expected:

- The policy is deployed as **Inactive** by default - activate it in **Setup → Transaction Security Policies**
- Ensure the referenced Apex class or Flow exists in your org
- Check that all configuration requirements listed in the policy details are met

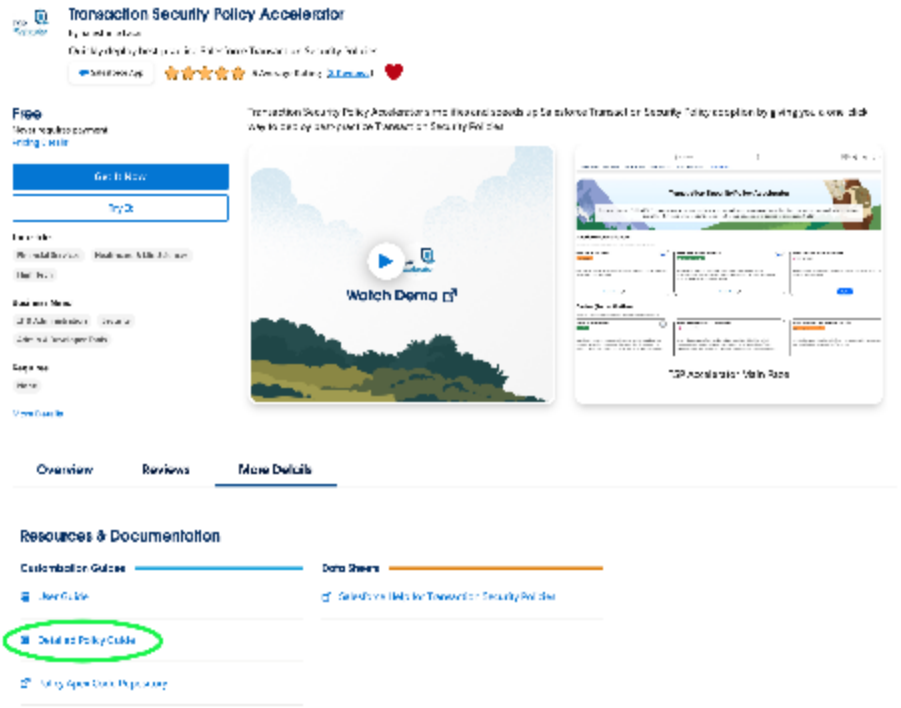
High Risk Data policies are not working
If a policy that detects high risk data (ex) is not functioning as expected, ensure you have marked the appropriate Data Sensitivity Levels as High-Risk in Setup and applied the Data Sensitivity Level to the appropriate field(s). Here is a walkthrough of how to complete these steps.

Detect API Access from Unapproved Third Party Apps is detecting all API Events
This policy is triggered whenever an ApiEvent is generated by a client application that is not included in the approved allowlist. To maintain the allowlist of approved clients, add custom metadata records to Approved_API_Client__mdt.
For each approved client, ensure that Approved_API_Client__mdt.Client_Name__c exactly matches the value of ApiEvent.Client. Because the match must be exact, you may want to query ApiEvent records to verify the correct client name before adding it to the allowlist.

Threshold for the Total Access and Export Detection Policies
As of v1.7, the 10,000 record threshold used across the Total Access and Export Detection for API Queries, Total Access and Export Detection for List Views, and Total Access and Export Detection for Report Exports policies is configurable. See the policy card for steps on how to set this up.

Understanding Policy Logic
If you need additional assistance understanding the logic behind each policy, you can review the Detailed Policy Guide available from the AppExchange listing. Furthermore, you can review the Apex

code for each policy in the open-source repository.



Customize Policies
If you would like to modify a policy to meet your specific business needs, you can use the Apex code for each policy that can be found in the open-source repository as a template that you then customize to fit your use case.

Large Volume of Records in trxsecurity__RecordAccessLog__c
Records in this object are created when you activate the TSP - Process Record Access Event Flow, and are used to log how many records a user is accessing. These records and figures are then used in logic for the Total Access and Export Detection for API Queries, Total Access and Export Detection for List Views, and Total Access and Export Detection for Report Exports policies. It is recommended that you activate the TSP - Delete Record Access Records Flow, which deletes trxsecurity__RecordAccessLog__c records that were created more than 2 days ago and are thus no longer needed for this use case. You may also create your own logic that calls the invocable method trxsecurity__DeleteOldRecordAccessLogs.deleteOldRecords(...) to delete these records.

Deployment failed - Status 400: Bad Request - You don't have sufficient access to perform this action
As the error message suggests, this is a permissions issue with the user authorized in the TSP_Accelerator Named Credential. Review that user's profile and permission sets to ensure they have the permissions needed to create Transaction Security Policies - this includes the new Modify Transaction Security Policies user permission that was recently added. See this Help article for more information.

📚 Resources

- Open Source Repository
- Interactive Walkthrough Demo
- Salesforce Transaction Security Policies Documentation
- Shield Platform Encryption
- Event Monitoring

📧 Support

For issues or questions, please share your thoughts in the Official: Shield and Security Center group on the Trailblazer Community.

🔑 Authentication Setup

⚠️ **CRITICAL**: You must configure the authentication chain (External Client App → Auth Provider → Named Credential) for policy deployment to work.

Step 1: Configure the External Client App

1. Navigate to **Setup** → **App Manager** → **New External Client App**
2. Under **Basic Information**, configure the following:
    - **External Client App Name**: `TSP_Accelerator`
    - **API Name**: `TSP_Accelerator`
    - **Contact Email**: `your email address`
3. Under API (Enable OAuth Settings), configure the following:
    - **Enable OAuth**: `check`
    - **Callback URL**: `https://YOUR-DOMAIN.my.salesforce.com/services/authcallback/TSP_Accelerator`
        - Replace `YOUR-DOMAIN` with your org's My Domain name
        - Your org's My Domain name can be found at *Setup → My Domain*
        - Examples:

- Production: `https://acme.my.salesforce.com/services/authcallback/TSP_Accelerator`
- Sandbox: `https://acme--dev.sandbox.my.salesforce.com/services/authcallback/TSP_Accelerator`
    - **OAuth Scopes**: Select the following:
        - `Manage user data via APIs (api)`
        - `Perform requests at any time (refresh_token, offline_access)`

4. Click **Create**
5. Navigate to **Settings** → **OAuth Settings** → **Consumer Key and Secret** and note down:

    - **Consumer Key** (you'll need this for the Auth Provider)
    - **Consumer Secret** (you'll need this for the Auth Provider)
6. **Wait 10 minutes** for these updates to propagate across Salesforce's infrastructure before they become active and can be referenced in the subsequent steps

Step 2: Create the Auth Provider
The Auth Provider connects your Named Credential to the External Client App for OAuth authentication.

1. Navigate to **Setup** → **Auth. Providers** → **New**
2. Configure the following:
    - **Provider Type**: `Salesforce`
    - **Name**: `TSP_Accelerator`
    - **URL Suffix**: `TSP_Accelerator`
    - **Consumer Key**: Paste the Consumer Key from Step 1
    - **Consumer Secret**: Paste the Consumer Secret from Step 1
    - **Authorize Endpoint URL**:
        - Production: `https://login.salesforce.com/services/oauth2/authorize`
        - Sandbox: `https://test.salesforce.com/services/oauth2/authorize`
    - **Token Endpoint URL**:
        - Production: `https://login.salesforce.com/services/oauth2/token`
        - Sandbox: `https://test.salesforce.com/services/oauth2/token`
    - **Default Scopes**: `api refresh_token`
3. Click **Save**
4. After saving, note the **Callback URL** displayed under **Salesforce Configuration** - it should match what you entered in the External Client App configured above

Step 3: Create the Named Credential
The Named Credential is what the Apex code uses to make authenticated Metadata API calls.

1. Navigate to **Setup** → **Named Credentials**
2. Click the **action button** next to *New*, then click **New Legacy**
3. Configure the following:
    - **Label**: `TSP_Accelerator`
    - **Name**: `TSP_Accelerator` (⚠️ MUST be exactly this - referenced in code)
    - **URL**: Your org's instance URL
        - Production: `https://YOUR-DOMAIN.my.salesforce.com`
        - Sandbox: `https://YOUR-DOMAIN.sandbox.my.salesforce.com`
    - **Identity Type**: `Named Principal`
    - **Authentication Protocol**: `OAuth 2.0`
    - **Authentication Provider**: `TSP_Accelerator` (Created above)
4. Click **Save**

5. Log into your org
6. Review the access request and if you approve, click **Allow**
7. Review the External Access request and if you approve, click **Confirm**
8. Validate that the **Authentication Status** shows as **Authenticated**

Authentication Status	**Authenticated as** epic.065065764fe3@orgfarm.salesforce.com

🔄 Version Control

Version	**Notes**
1.2	- Initial release, welcome!
1.3	- Additional Transaction Security Policies added - Fixed issue where 'Why Does It Matter' field was getting cut off
1.3.1	- Fixed a minor issue where the configuration mentioned for the Impossible Travel Detection policy did not represent what was needed
1.3.2	- Fixed issue with the Event Name for the Detect High Risk Data in Bulk API Result Downloads policy
1.3.3	- Fixed issue where the TSP Accelerator tab/page was not loading when a user had Lightning Component Debug Mode enabled
1.4	- Added the Detect Logins from Countries on Blocked List policy
1.4.1	- Fixed an issue where some policies when edited could cause a duplicate name error. Specifically, the following policies have been renamed: Total Access and Export Detection for API Queries -> Total Record Detection for API Queries Total Access and Export Detection for List Views -> Total Record Detection for List Views Total Access and Export Detection for Report Exports -> Total Record Detection for Report Exports - Fixed a

Version	Notes
	minor issue with the progress bar in the Configure and Deploy modal
1.5	- Added a batch job, which is also an invocable method (e.g. trxsecurity__DeleteOldRecordAccessLogs.deleteOldRecords(...)), to ensure large data volumes can be cleaned up with the TSP - Delete Record Access Records flow.
1.5.1	- Added the new “Modify Transaction Security Policies” user permission to the TSP Accelerator Permission Set
1.5.2	- Improved the efficiency of the query logic in the TrxSec_DetectTotalAccess classes to better handle large data volumes
1.6	- Fixed an issue where the Detect Report Exports by Unapproved Profiles and Total Record Detection for Report Exports policies did not recognize asynchronous report exports or exports through the Excel connector - The Total Record Detection for API Queries, Total Record Detection for List Views, and Total Record Detection for Report Exports policies now include the records from the current event when evaluating the 10,000 record threshold, so a single large query or export can trigger the policy - Improved the Detect High Risk Data in Bulk API Result Downloads policy to inspect subqueries within the SOQL and resolve child relationships, so high risk fields on related records are now detected - Updated the Detect Dormant User policy to look for any successful login within the last 90 days, and to no longer apply to new users that have no login history - Added a Delete permission check and user mode enforcement to trxsecurity__DeleteOldRecordAccessLogs, so the job now respects the access of the running user - Improved the error message shown in the Configure and Deploy modal when the TSP_Accelerator Named Credential is not authenticated or its URL is incorrect

Version	Notes
1.7	- The 10,000 record threshold used by the Total Record Detection for API Queries, Total Record Detection for List Views, and Total Record Detection for Report Exports policies is now configurable in Setup → Custom Metadata Types → Detection Threshold. Each policy is evaluated against its own threshold, and if more than one record exists for the same event the lowest value is used - Added the ability to exclude specific users from a subset of policies via the Policy Exclusion custom metadata type in Setup. Supported policies are Detect Impossible Travel Scenarios, Detect Login As Events, Detect API File Downloads, Total Record Detection for API Queries, Total Record Detection for List Views, and Total Record Detection for Report Exports - Improved scrolling in the Configure and Deploy modal so the height follows the policy content, up to 90% of the viewport
1.7.1	- Fixed an issue where the Monitor Internal Logins that Bypass SSO policy only detected standard password (Application) logins. The policy now also detects passwordless UI logins that bypass SSO, including Passwordless Login, Passwordless Login via Passkeys, and Lightning Login