IBM Security Secret Server

Privileged Behavior Analytics

IBM

Table of Contents

22 June 2020

Overview

Privileged Behavior Analytics works with your Secret Server to improve the security of your enterprise systems by helping to visualize, detect, interrupt, and announce threatening activity and behavior across your IT infrastructure.

- Visualize: by applying algorithms to Secret Server log files, Privileged Behavior Analytics visualizes data relationships to help your staff recognize and respond to security threats

- Detect: Privileged Behavior Analytics learns patterns of activity - 'behaviors' - associated with security threats and continuously monitors for such threat indicators

- Interrupt: by mounting Access Challenges of several types, Privileged Behavior Analytics automatically interrupts concerning behaviors

- Announce: as it detects possible threats, Privileged Behavior Analytics uses several contact methods to notify appropriate staff

ARCHITECTURE

Privileged Behavior Analytics uses the Amazon AWS Cloud and advanced algorithms to provide you with insights.
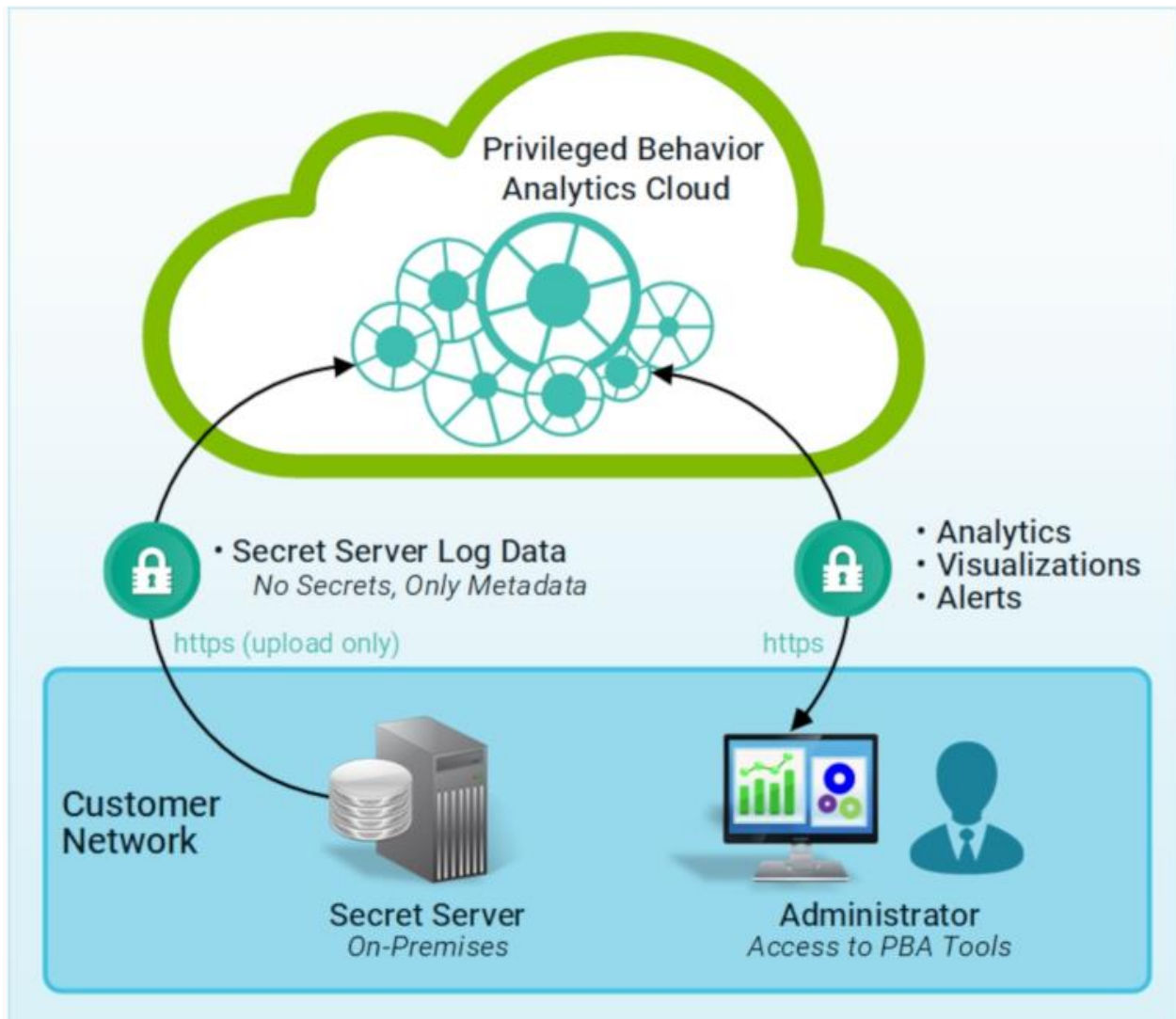
The following processes occur:

- Secret Server uploads activity logs to Privileged Behavior Analytics in the Cloud (AWS).
- Privileged Behavior Analytics applies advanced algorithms to the data to deliver alerts, analytics, and visualizations.
- To access these features, the administrative uses a web browser to authenticate with Privileged Behavior Analytics.

SECRET SERVER METADATA AND DATA SECURITY

Secret Server secures access to your company's most important resources. Privileged Behavior Analytics further secures those resources by generating insights about how your privileged users access the most protected resources. With training against a

suitable data set, Privileged Behavior Analytics alerts you when a privileged user is behaving in an irregular way, which might signal an intruder or an inside malefactor.

Privileged Behavior Analytics also secures those resources by adhering to stringent standards for data security.

Privileged Behavior Analytics resides in the cloud

As a Cloud service, Privileged Behavior Analytics is easily accessed and highly secure.

Secret Server uploads only metadata

After Secret Server is configured to work with Privileged Behavior Analytics, Secret Server uploads data securely to your organization's tenancy with the Privileged Behavior Analytics service. Privileged Behavior Analytics uses the event log data that is generated by Secret Server, so that only **metadata**—data about your data, not your data itself—go to the Cloud for analysis by Privileged Behavior Analytics.

This means that no actual Secret fields, such as passwords, private keys, notes, or other first-order data ever leaves Secret Server. Instead, only data about these things—literal **metadata**—is uploaded to Privileged Behavior Analytics.

For example, for a Secret that is a Windows account, the following data considerations occur:

- Fields that are uploaded for analysis by Privileged Behavior Analytics in the Cloud include the Secret Name, Secret Template, Secret Folder, Secret Policy ID, and Permissions. These are fields about the Secret, but not about the Windows account it contains.
- Fields that are not uploaded might include fields like Machine, Username, Password, Notes, Site, or any attached files, extra fields, or Secret keys as these comprise the actual content of the Secret.

Privileged Behavior Analytics users log in to the Privileged Behavior Analytics Cloud Service

Your organization's designated users securely log in to the Privileged Behavior Analytics service to use its analytics tools and configure alerts. Privileged Behavior Analytics continuously processes the log data and applies analytics to deliver insights and alert on anomalous behavior.

Data protections and security applied throughout

The design and build of Privileged Behavior Analytics always maintains the security of your Secret Server.

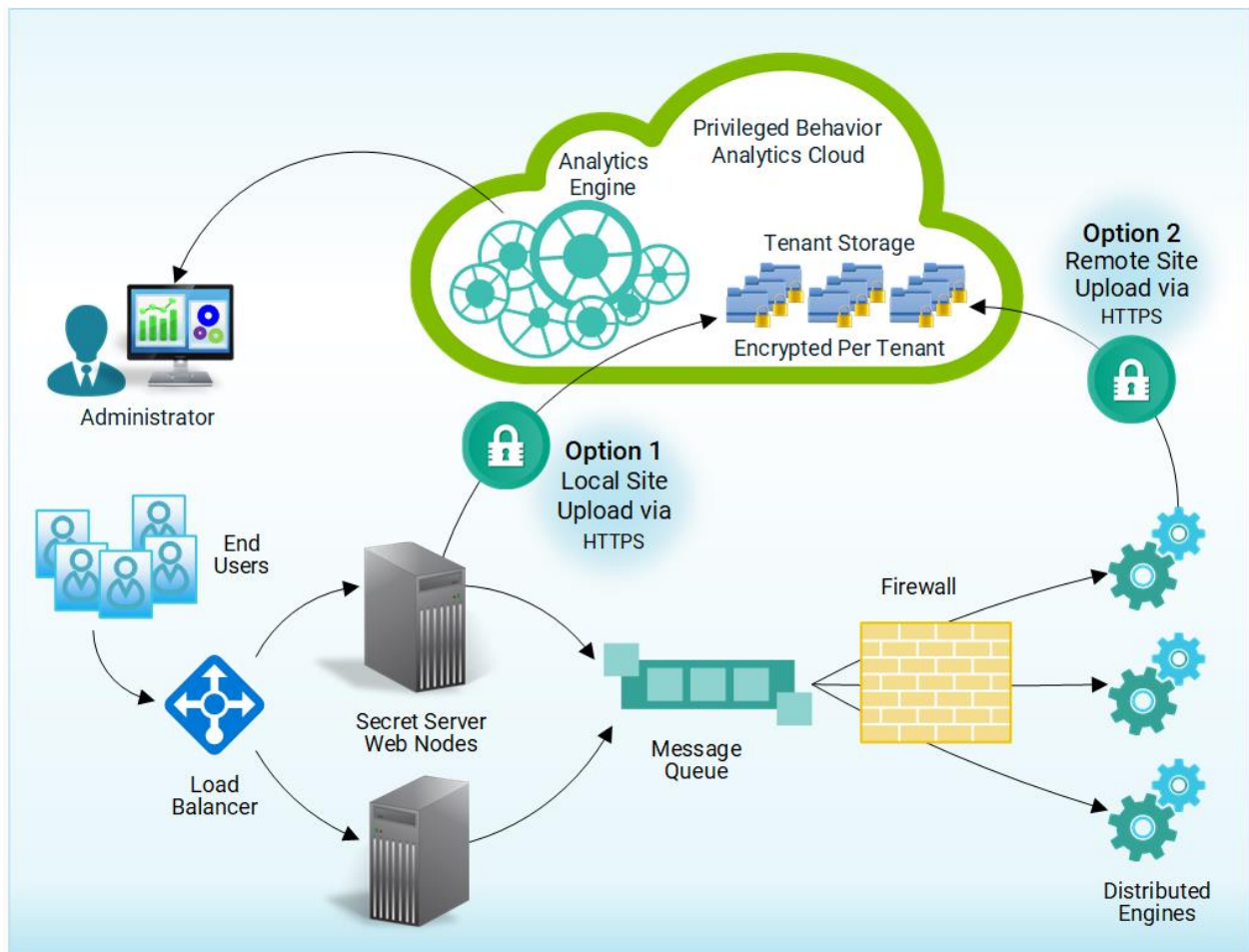## Security provisions apply to data uploads

Significant protection applies to data uploads by Secret Server to Privileged Behavior Analytics.

- Data sent to the cloud is through a one-way upload that can only be initiated by Secret Server.
- The data upload mechanism provides no means for remote access into your Secret Server.
- In addition, the data uploads to a Cloud location that only your organization's Secret Server can write.
- Privileged Behavior Analytics encrypts your organization's uploaded data with a key that is unique to your Privileged Behavior Analytics tenancy.

Alternative data upload by using a Distributed Engine Architecture

If your organization disallows outbound connections from Secret Server's network segment, you can upload by using the Distributed Engine architecture.

- Option 1 shows the default upload approach by the Secret Server web node in the local site.
- Option 2 shows a remote site upload by using a Distributed Engine.

Security provisions apply to connections to the cloud

To secure your data end-to-end, all connections to Privileged Behavior Analytics are encrypted with industry-standard Transport Layer Security (TLS) encryption. This includes all data uploads to the Cloud and all use of Privileged Behavior Analytics.

Built on Amazon Web Services

Privileged Behavior Analytics relies on the best-in-class security provided by Amazon Web Services. You can find out more about the underlying security of AWS at:

https://aws.amazon.com/security/

Strict access control and tenant isolation

Privileged Behavior Analytics features tenant isolation and exacting internal access controls that provide multiple safeguards against unauthorized access to any organization's data.

- Privileged Behavior Analytics isolates each organization's data from that of other organizations.
- Strong access controls give each Privileged Behavior Analytics operational component only the rights required to perform its role.

These layers of defense ensure that even were unauthorized parties to gain access to a part of the Privileged Behavior Analytics Cloud, their access would be isolated to that tenant and to the abilities of the compromised component.

Proactive monitoring

IBM continuously applies proactive monitoring protocols to the Privileged Behavior Analytics service.

- Administrators will receive alerts on any indication that someone might be trying to gain unauthorized access.
- Atypical behavior patterns among IBM's own administrative staff would likewise be flagged for review to guard against the emergence of inside malefactors.

Encryption at Rest

Privileged Behavior Analytics encrypts data at rest in the cloud. This additional layer of protection safeguards information in Privileged Behavior Analytics even should an unimaginable series of events somehow leave such data exposed.

VIDEOS

Click the following links to play a video demonstration.

Note: The video will not open in a new tab or window unless you use your browser's controls to tell it to do so. In many browsers, you can right-click and select a menu item for opening the link target in a new tab or new window. If you simply click the link, the video will play in this page. To return to this article, use your browser's previous-page tool (the 'back button').

- Alert Use Case

- Access Use Case

- Temporal Use Case

Getting started

Privileged Behavior Analytics resides on the Amazon AWS platform as a Cloud application. To use Privileged Behavior Analytics requires no hardware installation and no COTS installation on your premises.

However, it does require that you configure Secret Server to send metadata to Privileged Behavior Analytics, and this process is version dependent.

This articles in this section detail the required setup.

REQUIREMENTS

- Secret Server v10.5 or later or Secret Server SaaS

- Receipt of an email with your Privileged Behavior Analytics account login information

    - Provided you have purchased or been approved for a trial of Privileged Behavior Analytics

    - Single Sign On requires Secret Server v10.5 or later

SECRET SERVER CONFIGURATION

Secret Server provides data to Privileged Behavior Analytics through the Data Uploader, which requires a version-dependent configuration.

The following significant setup focus areas include:

- Configuring for proxied environments

- Import of historical data

- Setting up the Background Worker (clustered environments)

Data uploader setup

Version 10.5 and later, or SaaS editions

For Secret Server Version 10.5 and for Secret Server SaaS, event data is uploaded to Privileged Behavior Analytics via queues and micro-loading and is closer to real-time. Prior versions of Secret Server data upload followed the more typical data warehouse design of file upload and small batch-loading.

Single Sign On requires a key exchange for Privileged Behavior Analytics to use Secret Server as an identity provider, and a new integration key is provided with Privileged Behavior Analytics's public key in order to initiate this key exchange.

To obtain the Integration Key from Privileged Behavior Analytics that will be used by Secret Server to authenticate and upload data to Privileged Behavior Analytics, complete the following steps

1. Log into your Privileged Behavior Analytics instance and navigate to System Settings.

2. Click on View Integration Key.

3. Copy the key.
   If you are prompted to specify whether Secret Server is on version 10.4.000000, click Yes.

4. After copying the integration key, open Secret Server and navigate to Administration > Privileged Behavior Analytics.

5. Click Edit.

6. Enter your PBA key in the same-named field.

   a. The PBA Key field contains the secret access key and other parameters for uploading data to PBA.

   b. The key is encrypted for protection in transit; when you enter the key into Secret Server, it is encrypted and saved using your standard Secret Server encryption (AES-256, and DPAPI/HSM if configured).

   c. The key can never be loaded again through the UI, but can be updated in case the linked IBM PBA account must be changed.

7. Set the value for these fields:

   - **Enabled**: Enable Privileged Behavior Analytics.

- **Site:** Set to **Local** by default, this option will process and upload event logs by using your Internet Information Server running Secret Server (the default for your local site).
  If you are also using Distributed Engine you can specify a remote site and upload event data to PBA via Engines. This option is useful if you want to offload the work or if you prefer to allow an outbound firewall rule to the PBA servers from an Engine rather than from the server running Secret Server.

- **Challenge Enabled:** Enable Secret Server Access Challenges. See the Access Challenges article for further information. You may also click on Advanced to change additional settings.

- **External PBA URL**: This is the URL of your Privileged Behavior Analytics cloud instance. It is set automatically by the integration key but may be overridden. This URL is used for Single Sign On, for redirecting to PBA from the Tools menu, and on the Access Challenges page to create links to the PBA events that spawned Access Challenges.

- **Metadata Interval (Installed Only)**: The frequency that metadata is uploaded to PBA.
  The recommended interval is at least 60 minutes.
  The minimum interval is 5 minutes.
  Metadata frequency should vary based on how often new Users and Secrets are added in Secret Server; typically it should not need to be less than 60 minutes.
  For Cloud, this setting is unavailable and defaults to 60 minutes.

When the configuration is saved and PBA is set to enabled, the configuration is validated. It can also be manually validated by clicking **Test PBA Key**.

Proxied environments

If your Secret Server has outbound access through a proxy, its web.config must be modified to specify the proxy configuration.

If Secret Server is also clustered and has multiple worker roles enabled (see the Background Worker article), the web.config must be updated for each Secret Server in the cluster. Microsoft has more information on this.

The other option in a clustered environment is to specify a remote site for the data upload, and upload data through a Distributed Engine. If the distributed engine's host server is also behind a proxy, however, the engine's Thycotic.DistributedEngine.Service.exe.config must be modified similarly to the web.config in order to specify the proxy settings.

The web-proxy.config can be uncommented and updated to specify the proxy settings.
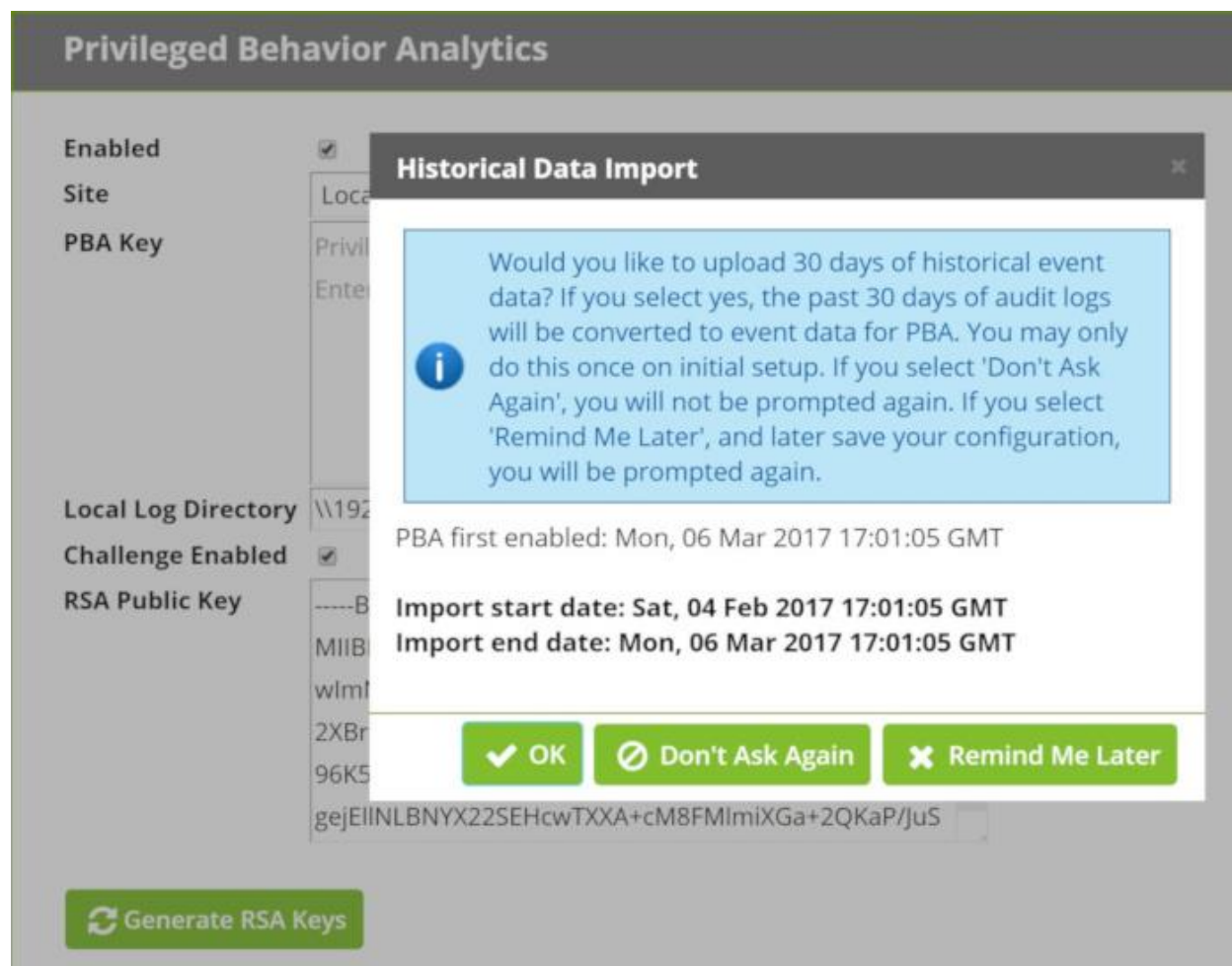
Historical data import

When you first enable Privileged Behavior Analytics, you will be prompted whether to import the last 30 days of event data.

Importing historical data reduces the learning period and potentially enables you to begin analyzing user behavior from day one, assuming Secret Server has been installed long enough to meet the learning period requirement.

Event data is not persisted in Secret Server, but audit data is persisted. Accordingly, Privileged Behavior Analytics derives events from audit data.

By default, only 30 days of historical event data is imported because this is typically the most relevant data for learning about user behavior.

If you seek to import more than 30 days of historical data, please contact IBM Support for assistance.

When you receive the prompt about whether to import historical data, select:

- **OK** to import 30 days of historical data

- **Don't Ask Again** to permanently dismiss the query

- **Remind Me Later** to receive the prompt again the next time you save a Privileged Behavior Analytics configuration with Privileged Behavior Analytics enabled.

If you select **Remind Me Later** and come back on a later date to import historical data, the original 30-day time range will be used for the import, that is, the 30 days immediately prior to the first date that you enabled Privileged Behavior Analytics. This ensures data continuity.

If you click **OK** to begin the import, you will see a dialog like the one below, stating the number of events to be imported and a time estimate for the import to complete. Immediately queued for processing, all the events must pass through the data pipeline for upload to Privileged Behavior Analytics.

Background Worker (Clustered environments)

After enabling Privileged Behavior Analytics, navigate to AdminClustering.aspx and ensure that at least one of your web nodes has the Background Worker column enabled.

The Background Worker feature allows you to specify the Secret Server nodes that run background tasks.

PRIVILEGED BEHAVIOR ANALYTICS CONFIGURATION

In Privileged Behavior Analytics, the User Settings allow password changes and configuration of per-user alert notifications. The System Settings allow the configuration of Secret Server integration, global alert and challenge callback, and time settings.

User Settings

You can navigate to User Settings by clicking the cogwheel symbol at the top right of any Privileged Behavior Analytics page and choosing User Settings.

Account Settings: Lets you change the password on your account used to access Privileged Behavior Analytics.

Alert Notification Settings: You can set the email address to receive alerts and specify whether you want to receive alerts and warnings as they occur.

System Settings

You can navigate to System Settings by clicking on the cogwheel symbol at the top right of any Privileged Behavior Analytics page and choosing System Settings.

Alert Threshold: The numerical value of an alert must meet or exceed to send an email.

Alert Action: Whether you wish to Challenge a Secret Server User if their actions cause Privileged Behavior Analytics to generate an alert for them that meets or exceeds the Alert Threshold. To use Challenges, you must configure it on Secret Server as well. More information on the configuration can be found in the following Access Challenges section.

Warn Threshold: The numerical value a warning needs to meet or exceed to send an email.

Warn Action: Whether you wish to Challenge a Secret Server User if their actions cause Privileged Behavior Analytics to generate a warning for them that meets or exceeds the Warn Threshold.

Secret Importance: Brings you to a page that lists all your Secrets and lets you change any of their importance settings in Privileged Behavior Analytics.

User Watch List: Check the boxes to automatically watchlist users with active alerts and warnings or new users. If the status of the user changes (for example, their active alert is cleared, or a new user reaches 30 days), then the user will be automatically removed from the User Watch List.

View Integration Key: This key is copied to Secret Server and provides access information for Secret Server to authenticate with and upload data to Privileged Behavior Analytics.

PBA Key Pair / Secret Server Key Pair: Key exchange is used by Privileged Behavior Analytics during Single Sign On in order to verify Secret Server's (as an identity provider) user claims. In the opposite direction, it is used by Secret Server as an additional layer of security to verify that Access Challenges were signed by the authorized Privileged Behavior Analytics instance.

Initiate Key Rotation: Privileged Behavior Analytics initiates a key rotation in which both Secret Server and Privileged Behavior Analytics generate a new key pair and exchange the new

public key with each other using the last public key to sign this new exchange. Keys are typically rotated periodically as a security best practice.

Clear Keys: This is used only when migrating from one Secret Server instance to a completely new Secret Server instance while using the same Privileged Behavior Analytics instance or when troubleshooting issues with key exchange. CAUTION: This clears all key pairs (both Secret Server and Privileged Behavior Analytics) from Privileged Behavior Analytics's database. After clearing, the integration key is copied to the target Secret Server and the initial key exchange is conducted, the same as with a fresh configuration of Privileged Behavior Analytics-Secret Server integration.

SINGLE SIGN-ON

Secret Server can act as an identity provider for Privileged Behavior Analytics.

- Any user with the View Security Analytics role permission in Secret Server may log into Privileged Behavior Analytics.

- Additionally, any user with Administer Security Analytics role permission can perform administrative actions once logged into Privileged Behavior Analytics through Single Sign-On (SSO).

- Local Privileged Behavior Analytics users (the initial users prior to integrating Privileged Behavior Analytics into Secret Server) still have administrative rights as well.

Typically, Single Sign On will start working without additional configuration.

Verify Single Sign-On

Verify that the Privileged Behavior Analytics and Secret Server key pairs both show a status of **Confirmed** on the following pages:

—`<SECRET SERVER>/AdminAnalyticsView.aspx` and

-`<PBA>/system_settings`

This key exchange is used for verification of Secret Server as an identity provider.
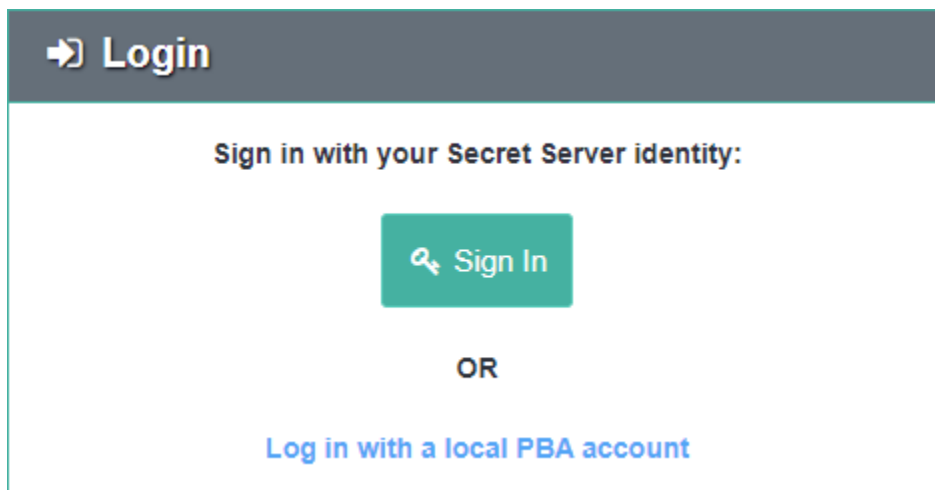
To verify that the SSO claim was signed by Secret Server, Privileged Behavior Analytics must have a copy of Secret Server's public key. Secret Server has infrastructure for key exchange and rotation between Secret Server and Privileged Behavior Analytics.

- When the integration key is first copied from Privileged Behavior Analytics and saved to Secret Server, it contains Privileged Behavior Analytics's initial public key.

- Secret Server then generates its own key pair and sends its public key to Privileged Behavior Analytics.

- Privileged Behavior Analytics registers Secret Server's public key and sends confirmation back to Secret Server.

When a key rotation is initiated, Privileged Behavior Analytics generates a new key pair and sends a signed request to Secret Server. The rest of the process is the same as the initial key exchange, except that each message is signed and verified during the rotation.

Troubleshooting

If Secret Server or Privileged Behavior Analytics shows that its Key Pair status is Pending Confirmation, try the Resend Confirmation button in either application.

- For example, if in Secret Server its key pair is **Pending**, then click **Resend Confirmation** in Privileged Behavior Analytics, so that Privileged Behavior Analytics will retry communicating to Secret Server that Privileged Behavior Analytics did register Secret Server's latest public key.

ACCESS CHALLENGES

Privileged Behavior Analytics can automatically issue Access Challenges on detection of anomalous behavior in Secret Server.

- Rule-based Challenges allow you to directly specify what qualifies as anomalous, for example, you can set a rule to issue a Challenge when a user's risk level exceeds a pre-determined threshold.

- When Privileged Behavior Analytics sends a Challenge to a Secret Server user, it can temporarily suspend the user's access to Secret Server. The user or an administrator must clear the Challenge for the user to regain access.

Privileged Behavior Analytics can challenge only users with the Secret Server Allow Access Challenge Role permission. This Role permission applies by default to all users except those with the Unlimited Administrator role permission.

Secret Server configuration for access challenges

Navigate to **<SECRET SERVER>/AdminAnalyticsEdit.aspx** and check **Challenge Enabled**.

Privileged Behavior Analytics

Enabled	☑
Site	Local ▼
PBA Integration Key	q59QXFBcb9A4oPQIpTuMgJ+vlDTlzKeDZ4nxDAQDFau QDZfXKi0esZBp0Gm5ML9r4dzGkaYslCVGyTNt3D+ubL5 ki5wTqftvfV0KOFk/sfDgfkDliIGkXXkNHV3A6K5zTaYhw3l C+p3swrr4t6Ydi5Pbc0beZTXiIg/rZ2Q6/ujTOaCD4hT97C T0GVc+IJZQflIZFvZVJ5qjC8Ro/Wk7ZHpjdlCY19AgH8L+7p R03mLrvFN5pG8w==
Challenge Enabled	☑
SAML Lockout	☑
Integrated Lockout	☑
Owner/Editor Requires Approval is Respected	❷ ☑

Additional settings

SAML Lockout: When this is set, authentication attempts via SAML can be blocked pursuant to a Lockout Challenge. If this setting is disabled, Lockout Challenges will apply only to non-SAML local authentication attempts where Secret Server is the identity provider. It is suggested that you enable this setting.

Integrated Lockout: When this is set, authentication attempts via Active Directory can be blocked pursuant to a Lockout Challenge. If this setting is disabled, Lockout Challenges will apply only to non-AD local authentication attempts where Secret Server is the identity provider. It is suggested that you enable this setting.

Respect Owner/Editor Requires Approval: This setting relates to Secrets subject to an **Owner Requires Approval** or **Editor Requires Approval** condition based on a global setting, a Secret Policy, or a setting in the Secret itself.

- If this setting is disabled, such **Requires Approval** conditions on a Secret *will not* be respected during a **Requires Approval Access Challenge**.
- If this setting is enabled, such conditions *will* be respected during such a challenge.
  - **Example:** A user is an editor of a Secret, and the effective setting on the Secret is *Editor Requires Approval=True*. However, the Privileged Behavior Analytics setting to **Respect Owner/Editor Requires Approval** is not active. Therefore, when a **Requires Approval Access Challenge** is processed for this user, the user will be able to access the Secret despite being subject in Secret Server to an *Editor Requires Approval=True* condition.
    - In this example, if the Privileged Behavior Analytics setting *was* active, the user must request approval to access the Secret until the Challenge is cleared, because Privileged Behavior Analytics respects the Secret Server *Editor Requires Approval=True* condition.
  - **Example:** A user is an editor of a Secret, and the effective setting on the Secret is *Editor Requires Approval=False*. However, the Privileged Behavior Analytics setting to **Respect Owner/Editor Requires Approval** settings is still not active. Therefore, when a **Requires Approval Access Challenge** is processed for this user, the user will *not* be able to access the Secret despite having in Secret Server an *Editor Requires Approval=False* condition, because Privileged Behavior Analytics is not set to honor that setting.
    - In this example, if the Privileged Behavior Analytics setting *was* active, the user would need to request approval to access the Secret until the Challenge is cleared, because Privileged Behavior Analytics respects the Secret Server *Editor Requires Approval=True* condition.

Privileged Behavior Analytics configuration for access challenges

Log into your Privileged Behavior Analytics instance and navigate to **System Settings** by clicking on the cogwheel symbol at the top right of any Privileged Behavior Analytics page and choosing **System Settings**.

Under **Global System Settings**, you can set triggers for Event Actions (an Alert or Warning) when conditions meet a certain risk threshold.

Currently, two simple rules may be set, a **Warning Threshold** and an **Alert Threshold**, with the Warning Threshold naturally lower than the Alert Threshold. For both, there are options to set up a Secret Server Challenge or run a Webhook or Codehook.

In setting up a Challenge, you must specify the Challenge **Type**.

Secret Server has the following Challenge Types:

- **Login:** the user must re-authenticate with Secret Server.
- **Two Factor**: the user must re-authenticate with Secret Server and the **Two Factor Remember Me** is expired (if set)
- **Require Approval**: the user must request approval for accessing any Secrets unless:
  - they are the only Approver for that Secret, *or*
  - they are the Owner or Editor; the Secret has *Editor/Owner Requires Approval* disabled; and PBA Configuration has *Respect Editor/Owner Requires Approval* enabled
- **Lockout**: the user is locked out from Secret Server until expiration of the Challenge or until it is cleared by an Admin user (a user having the **Administer Security Analytics** Role permission in Secret Server)
- **Record Session**: for Secrets that are capable of session recording the user will have their session recorded surreptitiously

You must also specify the Challenge **Duration** in minutes.

- If Duration is set to 0 minutes, the Challenge will never expire and the target Secret Server user will be denied access until they or an Admin clears the Challenge.

Access challenge administration

Challenges processed by Secret Server may be viewed and administered on **<SECRET SERVER>/AdminChallengeView.aspx**. This page is accessed by clicking the **Administer Challenges** button on **AdminAnalyticsView.aspx**.

- Challenges may be filtered by username and status (Cleared or Uncleared).
- If the External PBA URL is set, the **PBA Event Id** column will display a link to the Event Details page in Privileged Behavior Analytics for the Event that triggered the Challenge.

The following additional columns are displayed:

- **Cleared By**: This is the user (if any) that cleared the Challenge.
  - For a Login Challenge, all of a user's Secret Server sessions are ended and they must log into Secret Server successfully to clear the Challenge.
  - If a user does not have the **Allow Access Challenge** Role permissions in Secret Server, the Challenge will still be recorded, but will be listed as cleared by the **ThycoticSystem** user.
  - Finally, if an Administrator clears a Challenge on this page, that Administrator's username will be listed.
- **Type**: This is the Challenge Type as specified in the rule configured in PBA.
- **Start Date**: The time that an Event occurred in PBA and triggered the Challenge Event Action.
- **Cleared Date**: The date (if ever), the Challenge was cleared. If the Challenge has not been cleared, but Duration (in minutes) has passed, the Challenge will be listed as **Expired**.
- **Failure Count**: The number of times the user failed to clear the challenge, such as by failing to successfully authenticate.
- **Clear Button**: This is visible if you have the **Administer Security Analytics** Role permission in Secret Server. It allows you to clear a Challenge for another user.

Access challenge security
Because Access Challenge affects Secret Server user access from an external system, the architecture is heavily focused on security.

Privileged Behavior Analytics operations

After you set up Secret Server and Privileged Behavior Analytics to work together, you will begin normal operations. During a typical session with Privileged Behavior Analytics, you will use various tools, including:

- **Dashboard**: all your key indicators, neatly assembled

- **Privileged Behavior Alerts**: alerts issued by Privileged Behavior Analytics based on observed variances from typical Secret access and Admin action patterns, according to thresholds you set; includes current alerts and access to retired alerts

- **User Watch List**: assembles information about users whose activity accessing Secrets has attracted your specific scrutiny

- **Secret Event Clock**: visualizes the overall rate of Secret activity over time

- **Secret Event Graph**: visual representations of data about access to Secrets, designed to reveal patterns against which non-normative access will stand out

- **Secret Event IP Map**: visualizes Secret activity on a map, aggregated by IP address and location

- **Most Active Secrets**: reveals which Secrets have seen the most access and by whom

- **Most Active Users**: identifies users who are accessing more Secrets than most other users

- **Mobile Cache**: visualizes users that have performed a mobile cache on Secrets

- **Admin Actions**: like Secret Events, these pages include a Clock, Graph, IP Map, and Most Active Admins and Actions for administrator activity

- **Secret Details**: runs down all recent access activity for specific Secrets as well as the characteristics of the Secret

- **User Details**: allows you to explore in detail information that may be collected about a specific user's activity over time

- **IP Address Details**: for Secret Server instances that record various user IP addresses this shows IP activity trends and location information

DASHBOARD

Privileged Behavior Analytics's **Dashboard** landing page collects commonly used tools and views so you can easily recognize anything that would be out of the ordinary for your Secret Server environment.

- Multiple widgets present at-a-glance data visuals that cycle through views of activity for the last day, week, and month.

- The widgets activate or deactivate when you click the controls for each on the left side of the Dashboard.

- Cycling pauses on the data view for the last day, week, and month when you click on the Day, Week, and Month controls.

- Additional settings (dashboard theme, widget settings, cycle duration) are accessible through the three horizontal bars icon on the top left side.
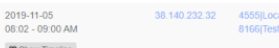
On the top right of the Dashboard you can click **Assistant** to activate the **Dashboard Assistant**.

- The Dashboard Assistant focuses you on recent events and allows you direct access to event records by clicking the title.

- The downward facing arrow icons on each event act to expand the record area to show you more information.

  - The extra information includes guidance on why the event has significance and what steps you should consider.

- Clicking the clockface icon at the bottom left of the Assistant application panel will open the **Assistant Archive** page and show past events.

PRIVILEGED BEHAVIOR ALERTS

The Privileged Behavior Alerts page (**Alerts** > **Privileged Behavior Alerts**) displays events in Secret Server that do not align with normally observed behaviors. This includes the capacity to display Alerts for circumstances you define (see the PBA Administration article).

Use the search field to locate specific Alerts by searching on text from any of the rows in the table. Columns include:

- **Severity**: whether the event justified an Alert (serious event) or a Warning (minor event)

- **Score**: the numerical score given to the event depending on its severity and the severity of incorporated events

- **User**: the Secret Server User who caused the Alert; clicking their name opens the **User Details** page

- **Range of Activity**: the time span within which the Alert occurred; includes an optional timeline graphic

- **IP Addresses**: the IP addresses used during the alert period with links to each IP Details page

- **Secret Accesses**: any Secrets accessed during the time span of the Alert that contributed to the Alert; clicking on the Secrets opens the **Secret Details** page

- **Admin Actions**: any administrative actions taken in Secret Server during the time span of the Alert that may have contributed to the Alert; clicking on the Admin Actions listed displays the table of all administrative activity for that User

- **Temporal Behavior**: a time entry will be listed here if the Alert occurred at a time the User does not normally access the Secrets involved in the Alert; clicking on the time entry will display the User's Temporal Data.

- **Actions**: button links open the **Alert Details** page, **Dismiss** the alert as normal behavior, or **Clear & Watch** the alert as abnormal behavior

    o To further investigate the Alert, log actions you have taken on the Alert, adjust the importance of any involved Secrets, or provide feedback to Thycotic on the usefulness of the Alert, click **Details**

o Clicking **Dismiss** or **Clear & Watch** will remove the alert from the page and save
it to **Historical Behavior Alerts**

Historical Behavior Alerts

The Historical Behavior Alerts page archives Alerts after they have been cleared from an Active state.

You can reach the Historical Behavior Alerts by navigating to **Alerts** > **Historical Behavior Alerts**.

In viewing Historical Behavior Alerts, note these fields:

- **Changed by**: the PBA User who cleared the Alert

- **Notes**: notes left on the Alert before it was cleared

USER WATCH LIST

The **User Watch List** page provides a convenient location to track users of interest and easily access information about each.

By default, the Privileged Behavior Analytics (PBA) System adds to the Watch List new users and those with active Alerts and Warnings.

Upon clearing Alerts and Warnings, or when a new User has been active for 30 days, the System removes them from the Watch List. These automated actions can be disabled from the **System Settings** page (see PBA Administration for more information).

On the right side of each User's Watch List entry are buttons to edit (reasons and notes) or delete the entry. For Secret Server customers that have a Secret Server Custom URL, an additional lock icon will appear, which links to the User's Edit page in Secret Server.

To add a Secret Server URL for direct linking:

- in Secret Server, hover over the Admin button in the toolbar

- select Configuration > Edit

- set the **Secret Server Custom URL**—the URL will be passed to Privileged Behavior Analytics in the next metadata upload

Users can be added to the Watch List by clicking the Add User icon at the top right of the page. Multiple new or existing users can be added to the list along with a list of reasons and notes.

- For existing Watch List users, the reasons and notes will be appended to their current reasons list and notes.

To make changes to a reason, click the **Edit Reasons** button on the top right side of the toolbar. The current list of reasons will appear with options to change the name or color or to delete the reason from the Watch List. These changes affect all Watch Listed Users with the edited reason and are not tracked in the Last Updated information.

SECRET EVENT CLOCK

The **Secret Event Clock**, in the Analytics section of Privileged Behavior Analytics, provides a temporal overview of Secret Server activity. It visualizes the distribution and concentration of activities over time for a given time range.

The coloring of the graphs range from white to dark blue.

- White means no activity.

- Dark blue means a lot of activity.

The center of the circular chart displays the number of events represented and the date range when they occurred.

- You can filter the temporal data by searching in the three boxes on the left for a:

    o Secret, Folder, or Secret Importance Level

- o   User, Account Type, Group, Name, or User ID

- o   IP address, City, Region, or Country

- This will refresh the graphs to reflect only events within the data range that are related to that Secret, User, or IP.

- If you refine by a Secret and wish to see which Users accessed that Secret on a day or at a certain time, you can right-click on the corresponding bar in any of the graphs and then click on the name of the Secret.

- Likewise, if you refine by a User and wish to see which Secrets they accessed on a day or at a certain time, you can right-click on the corresponding bar and click on the name of the User.

To move back and forth through specific weeks, use the left and right arrows at the base of the circular chart. If you wish to hide the side and bottom bar charts from the display, you can click the gray chart button to the top right of the circular chart.

SECRET EVENT GRAPH

The Secret Event Graph can be used to explore the behaviors of Secret Server users at a glance.

The graph is animated by default. You can pause the animation by clicking on the pause button at the top right of the graph area. Above the pause button is another button that will allow you to expand or collapse all nodes with a single click. You can also expand individual clusters by double clicking on them.

Each initial node circle, or Community, is a collection of users who are accessing similar Secrets. The larger the Community, the more users and Secrets there are inside it.

Communities may also have lines, or links, connecting them to other Communities. The links are an indication that a User or users within a Community are accessing Secrets that exist in another Community, indicating a possibility of accesses outside a User's responsibility. Thicker links represent more accesses between Communities.

When you expand (double-click) a Community, you can see all the users and Secrets it contains. The size of each node indicates how many accesses it has, and the thickness of links follows the same principal.

Community, User, and Secret nodes may be outlined by a shade of red. If this is the case, there is an active alert for a User and/or Secret and more information can be found by right clicking on the affected nodes and observing active alerts or by navigating to the **Alerts > Privileged Behavior Alerts** page.
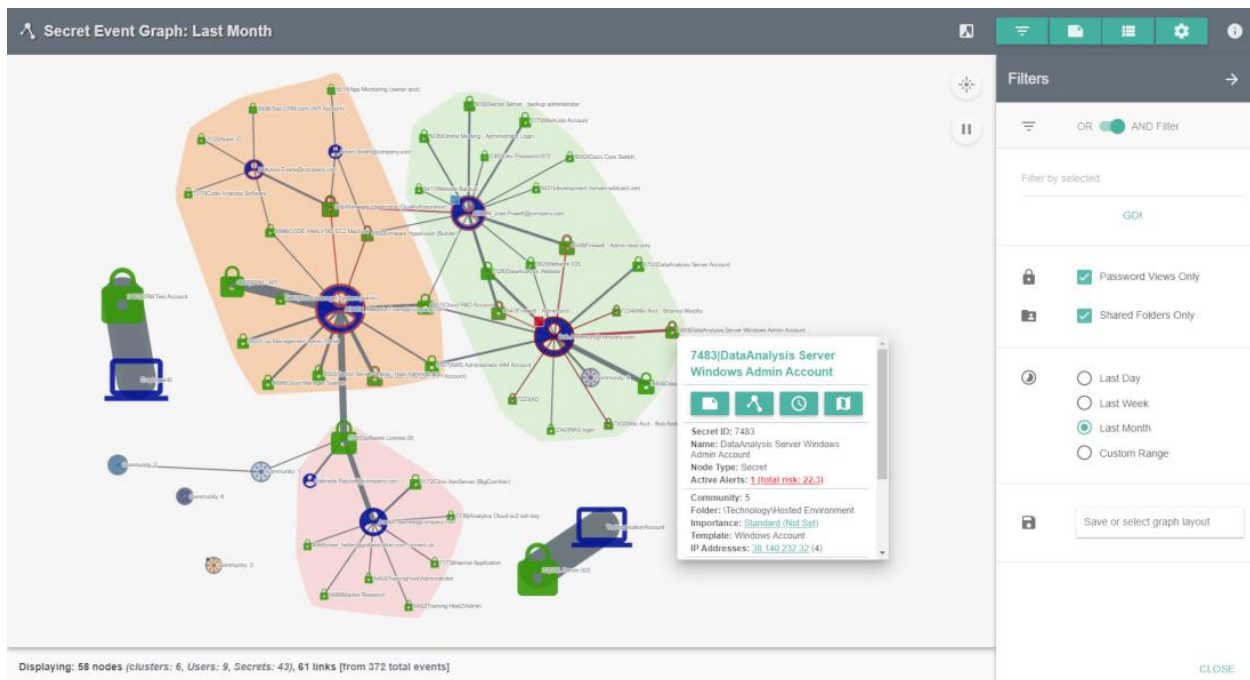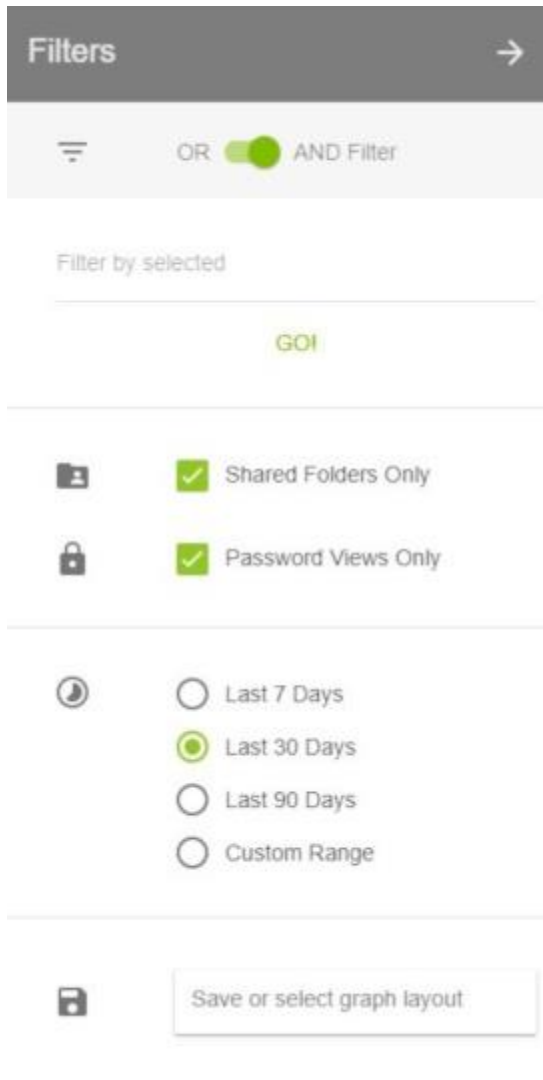
Whether a Community is expanded or not, you can right click on any node or link on the Secret Event Graph to add notes or see further details.

The icon to the left of the menu buttons will toggle the Graph between light and dark themes.

Filters

The Filters menu (three horizontal lines button) provides options to limit the number of nodes and links displayed.

- **OR AND Filter**: determines how filters will be applied to the Secret Event Graph

- **Filter by selected**: lets you filter the Secret Event Graph display by Secret, User, Group, Folder, IP Address, Secret Importance, and Template

- **Shared Folders Only**: unchecking this box will add Secret access activity from users' Personal Folders in Secret Server

- **Password Views Only**: turned on by default, this shows only Secret accesses, which include: web launches, passwords displayed, passwords copied to clipboard, Secrets edited, and Secrets exported; if turned off, all other Secret activity will be shown

- **Time Ranges**: by default, the Secret Event Graph will show activity from the last week; the Custom Range option allows selecting a start and end date to refine activity displayed

- **Save or select graph layout**: you may choose to save filtered views of the Secret Event Graph to quickly recall significant access landscapes

Notes

The Notes menu can be accessed by clicking the green button depicting a note with a folded corner at the top right of the Secret Event Graph page.
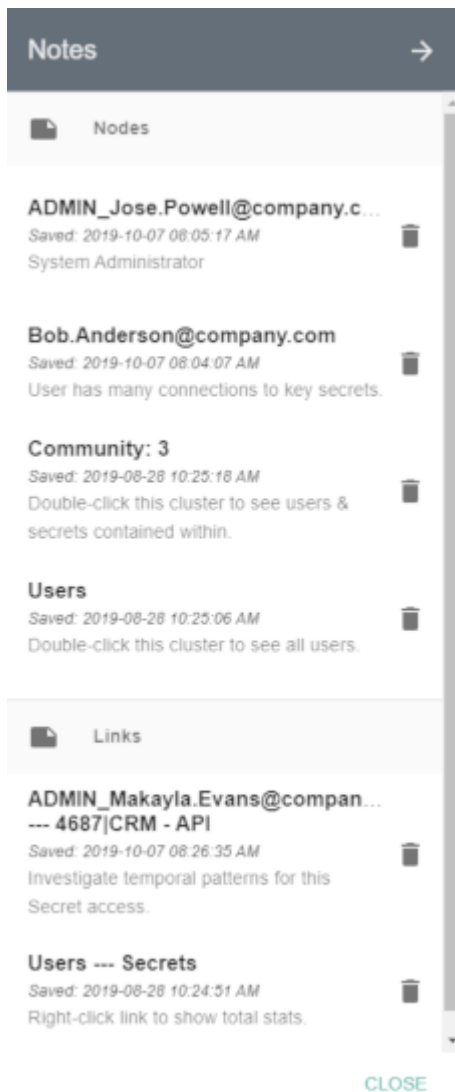
- All notes on nodes and links are listed here. You can edit any note by clicking on it or delete a note by clicking on the trashcan icon to the right of the note.

- Notes can be created by right-clicking on a node (circle) or link (line) in the Secret Event Graph. A small square of the color selected will appear on the node or link after the note is created.

- Hovering over the square or a note in the Notes menu will briefly highlight the note square on the Graph.

Table

The Table menu can be accessed by clicking the green button between the Notes and Tools buttons at the top right of the Secret Event Graph page.

This menu gives you a full, sortable text-based list of all User and Secret node metrics. Placing your mouse over any of the node names in the lists will highlight that node on the Secret Event Graph if the Community it is in is expanded.

- **Community** lists User and Secrets nodes and the Community number they are in

- **Secret/User** lists User and Secrets node names and whether each is a User or Secret

- **Number Connections** lists User and Secret nodes and how many accesses they have had or performed on other nodes

- **Number Unique** lists User and Secret nodes and how many unique Secrets or users, respectively, they are connected to

- **Last Active** lists User and Secret nodes and the timestamp of the last activity each had

- **First Active** lists User and Secret nodes and the timestamp of the first activity each had recorded in Privileged Behavior Analytics

- **Social Network Metrics** lists User and Secret nodes and the numerical value of the selected metric

Tools

The Tools menu (cogwheel button) allows you to customize what is displayed on the graph.
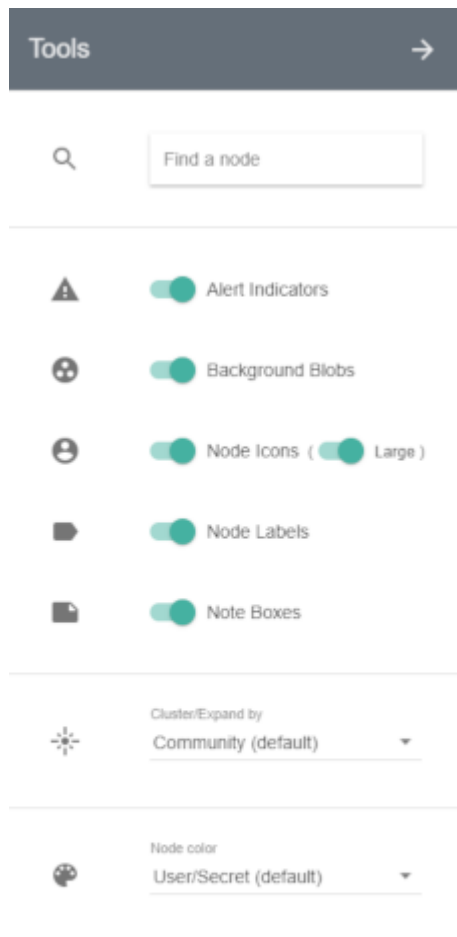
- **Search**: At the top of the menu is a search field where you can enter the name of a User or Secret to highlight that specific node on the Secret Event Graph. Press Enter to repeat the animation.

- **Background Blobs**: turned on by default, these surround all nodes in an expanded Community with a color similar to that of the collapsed Community

- **Node Icons**: turned on by default, this shows icons in place of circles for each user or Secret node. The size of the icon can be changed using the **Large** switch.

- **Node Labels**: turned on by default, these are Community numbers, Secret names, and User names shown next to each node

- **Note Boxes**: turned on by default, these represent notes that have been placed on any nodes or links

- **Cluster/Expand by**: by default, all nodes will be clustered by Communities; you can select the dropdown here to choose to cluster nodes by Secrets and users

- **Node Color**: there are multiple options for choosing how the nodes within an expanded Community are colored:

    - **Community**: all Secret and User nodes will be the color of the Community when it is collapsed

    - **Secret/User**: the User nodes are colored blue and Secret nodes are colored green (default coloring)

    - **Number Connections**: Secret and User node colors will range from white to red; the redder a node is, the more active it is

    - **Number Unique**: Secret nodes will always be white; User nodes will range from white to red, and the redder a node is, the more unique accesses it has

    - **Social Network Metrics**: these options can reveal important Secrets or users in the network

Table	→

Data Type
≡ Number Connections ▾

Node	↓ Number Connect...
Chase.McVicar@compa...	245
Jose.Powell@company....	213
Makayla.Evans@compa...	167
Caroline.Ward@compan...	150
Joseph.Barnes@compa...	149
1939|Product Demo Sys...	143
5040|Online Meeting 1 –...	141
5790|Software License 05	133
6250|Training Creator So...	111
Brianna.Murphy@compa...	111
Charles.Cooper@compa...	103
4687|CRM – API	88
Adam.Smith@company....	82
6506|Test.CRM.com (A....	80
919|Product Demo Syst...	68

SECRET EVENT IP MAP

The **Secret Event IP Map** summarizes Secret Server activity by IP address and location. This is useful for Secret Servers that allow access from external IP addresses.

Summary views and information provide a high-level understanding of recent and historical Secret events. Any anomalous locations in the data can be quickly observed, analyzed, and acted upon.

Map Key

The map shows by default the last week of IP address counts (purple circles) and active alerts (red triangles). You can click these features for further information.

Circle features sometimes have a gray or black outline, which means they are located only to the region (state) or country level, respectively.

Map Navigation

The Map can be navigated like most web-based maps:

- Pan to different locations by clicking on an open area and dragging the mouse

- Zoom in or out using the mouse wheel, the buttons on the upper left, double-clicking (+ Shift), or dragging the mouse while holding Shift to select a zoom box

*NOTE: If no data appear, please go to the **Filters** menu and try turning off filters or expanding the time range. From the System Settings page, a default location can be entered for cases where only internal (private) IP addresses are present in the data.*

Filters

The **Filters** menu (first menu button) provides options to limit data displayed based on user, Secret, IP address, location, and several related attributes.

- **OR AND Filter**: determines whether selected filters will be considered separately (OR) or together (AND)

- **Filter by selected**: filters the Map display by Secret, user, group, folder, IP Address, location, Secret importance, and template

- **Shared Folders Only**: unchecking this box will add Secret activity from users' personal folders in Secret Server

- **Password Views Only**: turned on by default, this shows only Secret accesses, which include: web launches, passwords displayed, passwords copied to clipboard, Secrets edited, and Secrets exported; if turned off, all other Secret activity will be shown

- **Time Ranges**: by default, the Map will show activity from the last week; the Custom Range option allows selecting a start and end date to refine activity displayed

- **Save or select IP map options**: save filtered views of the Map to quickly recall significant events or complex filter combinations
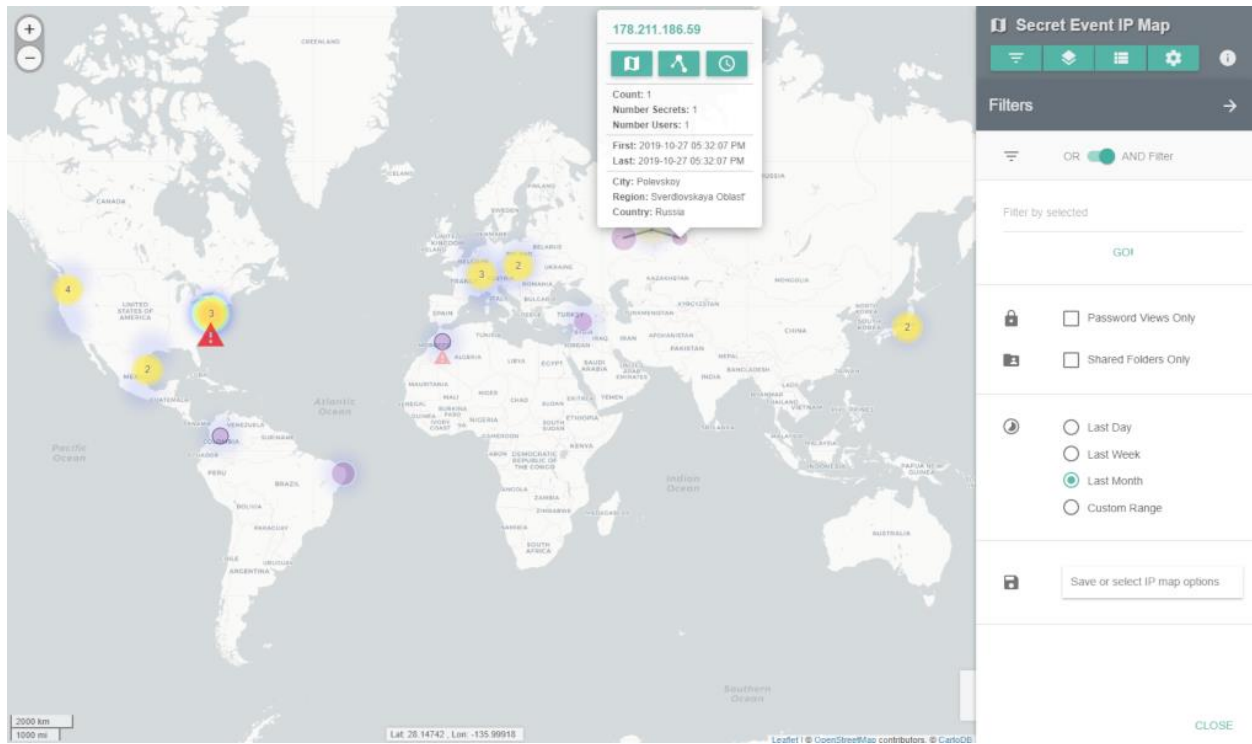

Layers

The **Layers** menu (second menu button) contains options to show circle layers as clusters (aggregated nearby points) or as individual points for the number of events, Secrets, or users active at each IP address.

There are also heatmaps and different basemaps that can be selected.

- Selecting the **Dark** basemap will change the entire map application to the dark theme.
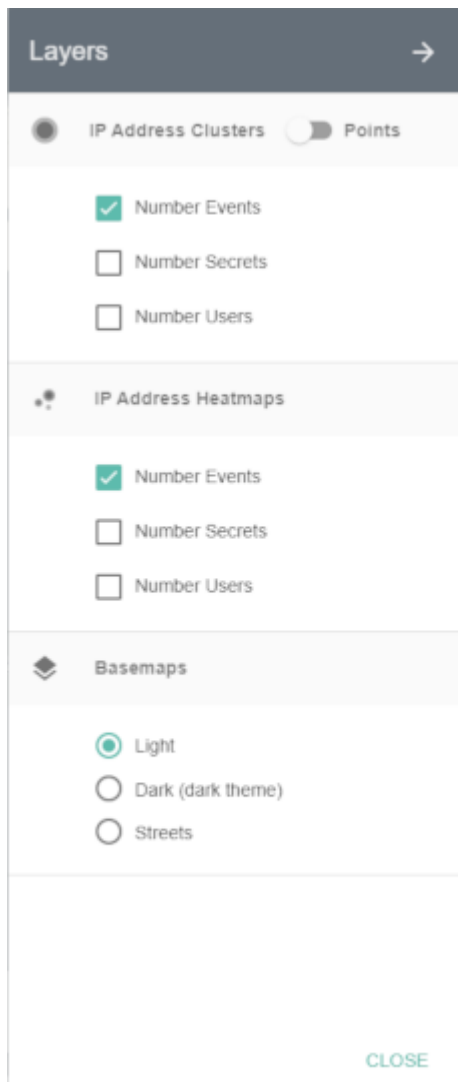
Table

The Table menu (third menu button) provides a sortable text-based list of key metrics (below) related to IP addresses. Placing your mouse over any of the rows in the lists will highlight a point or country on the Map.

- **Country – Number Events**: shows the total number of IP events within each country

- **IP – Number Events**: shows list of IP addresses with the total number of events for each

- **IP – Number Secrets**: shows list of IP addresses with the total number of Secrets accessed or modified from each

- **IP – Number Users**: shows list of IP addresses with the total number of users for each

- **IP – First Active**: shows IP address list with date of first activity observed

- **IP – Last Active**: shows IP address list with date of last activity observed

- **IP – City**: shows IP address list with city of the location (if available)

- **IP – Region**: shows IP address list with region (state) of the location (if available)

- **IP – Country**: shows IP address list with country of the location

Table	→

Data Type
Country - Number ...   ▼

| Country	↓ Number Events
United States of America	4,129
Italy	232
Poland	174
United States	154
Slovakia	153
Switzerland	60
Japan	27
Russia	22
The Netherlands	21
Brazil	16
Turkey	6
Colombia	2

CLOSE

Tools

The **Tools** menu (fourth menu button) provides a search function to find a specific IP address, city, region (state), or country among the data points currently loaded to the Map. Clicking on a result will pop up details and re-zoom the map to the selected IP address.

The Tools menu also contains two additional data layer options:

- **Alert Indicators** show red triangles for IP addresses that have active alerts or warnings. Clicking on a triangle will show details on the number of alerts and total risk score.

- **Country Counts** show a semi-transparent layer shaded by the number of events taking place in each country. Clicking on a country will re-zoom the map to the selected country, and hovering the mouse over it will show the country name, flag, and number of events.
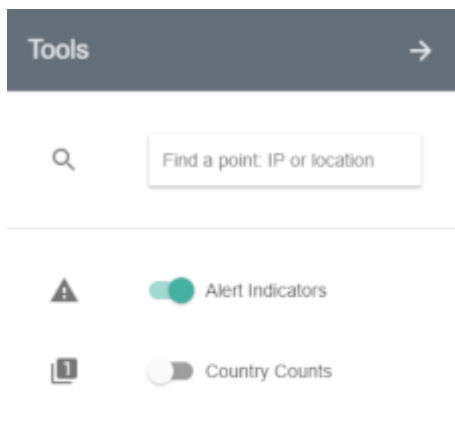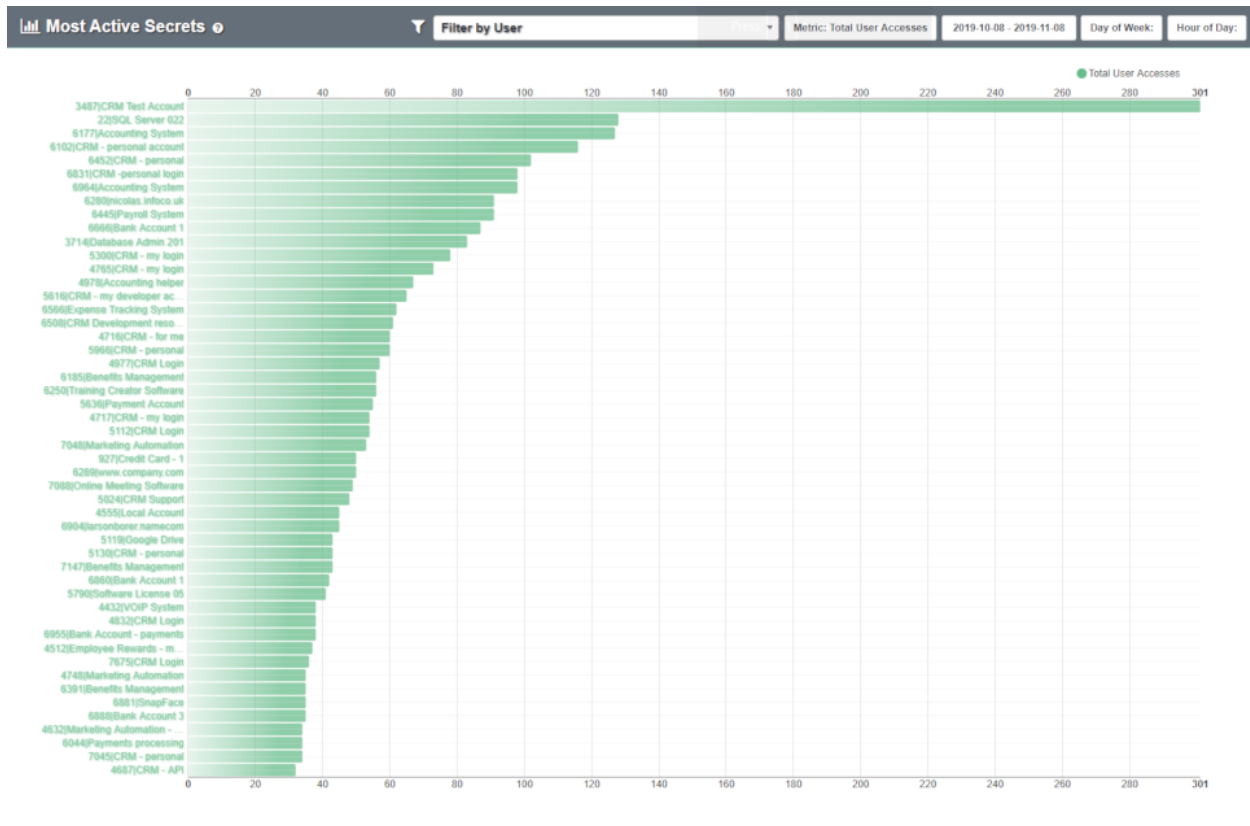

Secret Event IP Map Info

Next to the menu buttons is an info icon that launches the **Secret Event IP Map Info** box, which contains basic statistics on the map data displayed, map instructions, and disclaimers.

MOST ACTIVE SECRETS

Most Active Secrets ranks the top 50 most accessed Secrets. To see this, navigate to **Analytics** > **Most Active Secrets**.
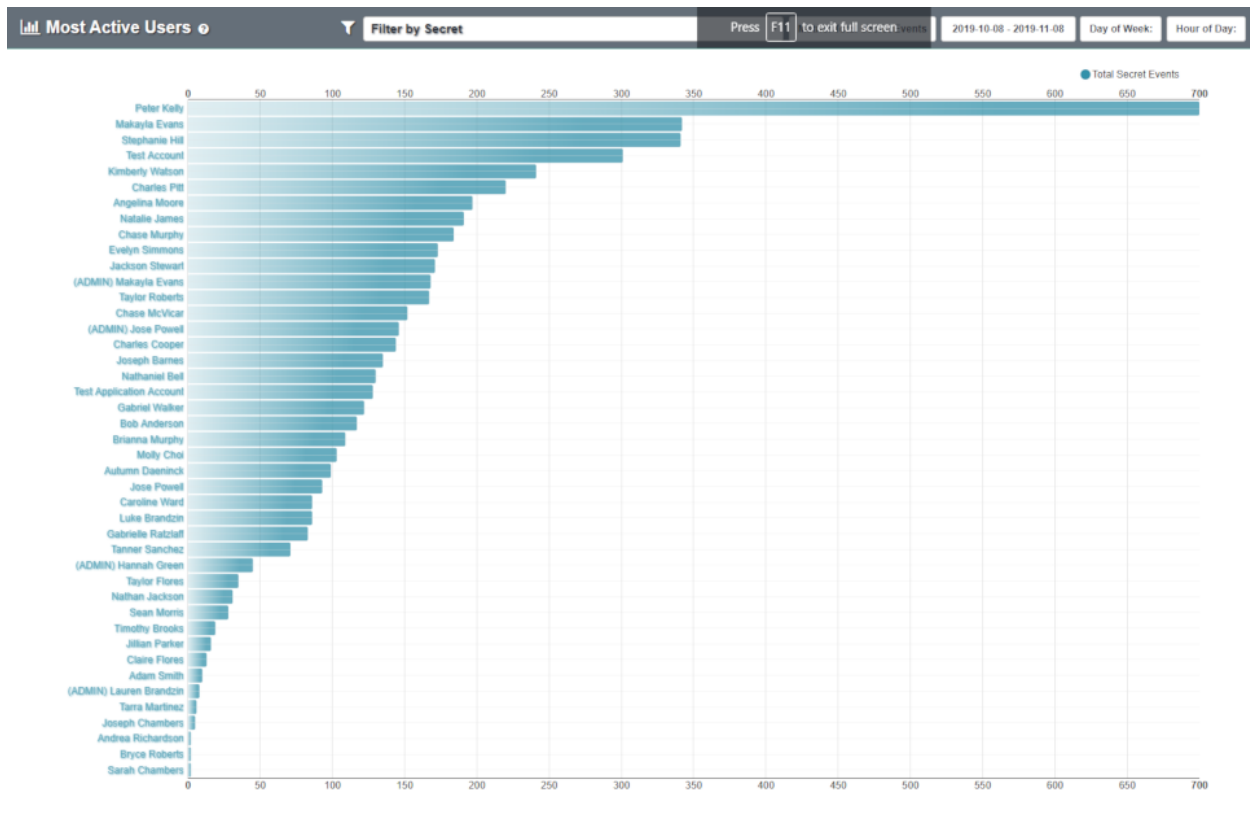
The list contains the Secret ID, Secret Name, and number of access events for each of the 50 secrets, using a bar chart as visual reference.

- By default, you will see the top 50 Secrets in your Secret Server environment for the past month.

- You can further filter the list by a User or specific timeframe.

- Clicking on a Secret in the list will take you to its **Secret Details** page.

MOST ACTIVE USERS

Most Active Users ranks the top 50 most active Users in your Secret Server environment. You can see your top 50 most active users by navigating to **Analytics** > **Most Active Users**.

The list contains the User Name, Display Name, and number of Accesses to Secrets, with a bar chart as visual reference.

- By default, you will see the top 50 Users in your Secret Server environment for the past month.

- You can further filter the list by a Secret or specific timeframe.

- Clicking on a User in the list will take you to the **User Details** page for that User.

MOBILE CACHE

Mobile Cache shows timelines for available cached Secrets organized by user. If any users have currently cached Secrets available from a Thycotic mobile or desktop application, it will be shown here.

You can view mobile-cached Secrets by navigating to **Analytics > Mobile Cache**.
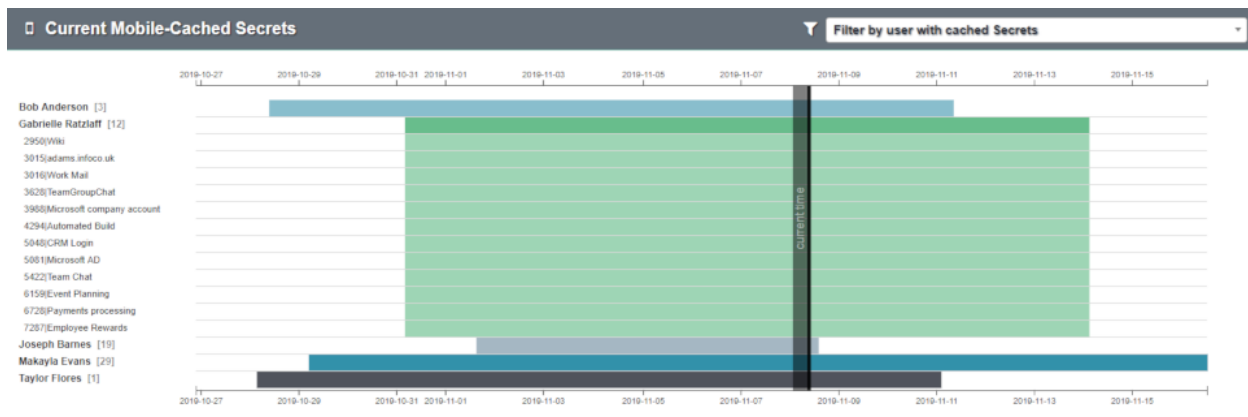
If multiple users have current mobile-cached Secrets, the filter at the top can limit the chart view to a specific user.

If any users have current mobile-cached Secrets, the timeline of the Secrets' availability is shown in the chart.

- Clicking on a user's bar will expand the view and reveal Secret names.

- Clicking the top bar again will collapse the Secret bars.

- You can drag the chart to pan left or right or scroll up or down to zoom the view.

- Scroll the entire page up or down from the right side of the window.

- Clicking on any of the user or Secret names will launch the details page.

ADMIN ACTIONS

Privileged Behavior Analytics also contains a suite of analytics based on Administrator Actions in Secret Server. Navigate to **Analytics > Admin Actions** to view the available pages.
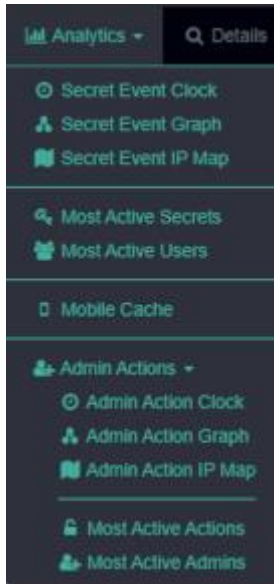
Like the Secret Event Clock, Graph, and IP Map, there are pages that show admins and Secret Server Actions instead of users and Secrets. Likewise, the **Most Active Actions** and **Most Active Admins** pages allow you to filter down the top 50 of each and easily access details pages.

SECRET DETAILS

The **Secret Details** page can be used to investigate how a Secret is being accessed from the perspective of many types of data collected on it. You can access Secret Details by navigating to **Details** > **Secrets**.

The Secret Details page lists all Secrets with Secret ID, Secret Name, Secret Template, the total number of events (Secret accesses plus modifications), number of Distinct Users that have accessed the Secret, the Created date, and the date of Last Activity.

If you click on any of the Secret names you will be directed to that Secret's Details page, which shows the following:

- **User Activity**: lists the most recent 500 encrypted accesses for the Secret, when they occurred, who accessed it, and how it was accessed.
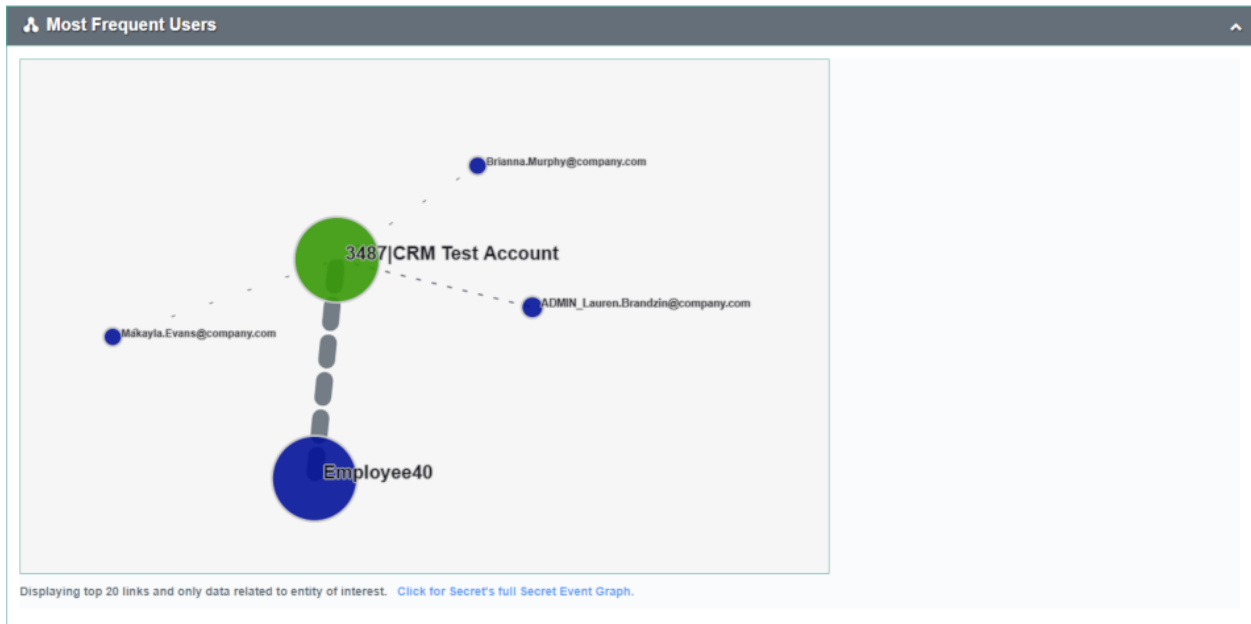

- **Activity Timeline**: shows all activity for the Secret, including alerts and warnings, accesses, and modifications as well as timestamps, IP address, and event details.
    - mouse over a colored circle for details on an event
    - pan left and right by dragging or zooming by scrolling, which also filters data in the table

- **Most Frequent Users**: an animated representation of the top 20 users accessing the Secret the most; you can zoom into the graph by scrolling or right-click on any node or link to view more details

Most Frequent Users

Displaying top 20 links and only data related to entity of interest. Click for Secret's full Secret Event Graph.

- **Temporal Behavior**: a chart showing all temporal data for the Secret organized by time of day and day of the week

  - the numbers across the bottom indicate the total events involving that Secret for that time of day

  - the values across the right side indicate the number of events involving that Secret for that day of the week

  - the legend at the bottom shows the number of events that correlate to the coloring of the chart blocks

    - mouse over a block to get the total number of events for that day of week and hour of day

Temporal Behavior

Total number events by hour and day of the week. Click for Secret's full Secret Event Clock.
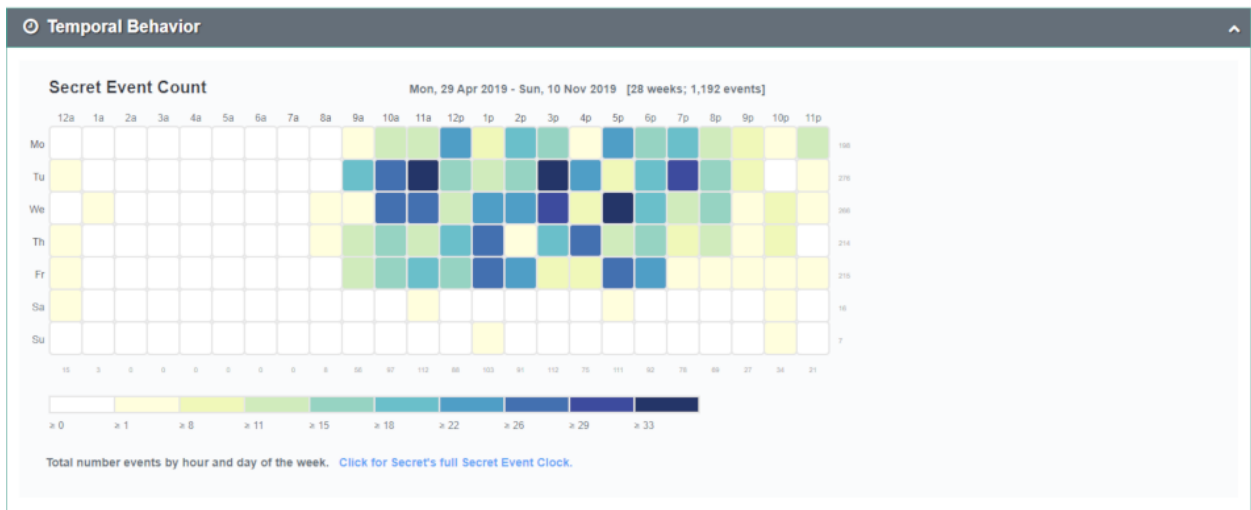
- **Secret Name and Folder History**: lists any changes that have been made to the name of the Secret or the folder it is kept in inside Secret Server

USER DETAILS

The **Active Users** page lists all Users, their Display Names, Account Type, total number of times they have accessed or modified Secrets, number of unique Secrets they have accessed, total number of administrative actions they have performed, when they were first seen in Privileged Behavior Analytics, and when they were last active.

The **User Details** page is the ideal place to dive deeper into a specific User's behavior from the perspective of many types of data collected on them. To see a user's details page, click on the user's name in the list of users.

Several sections display a variety of detailed information.

Activity Timeline: a chart showing when a User has performed Secret accesses, Secret modifications, or administrative actions in Secret Server, or has logged in or out of Secret Server over time

- each activity is denoted by a symbol shown in the legend at the top

- placing your mouse over any of the symbols in the graph will give more details on what the user did at that time

- grabbing and moving the side buttons on the bottom chart will zoom the top chart
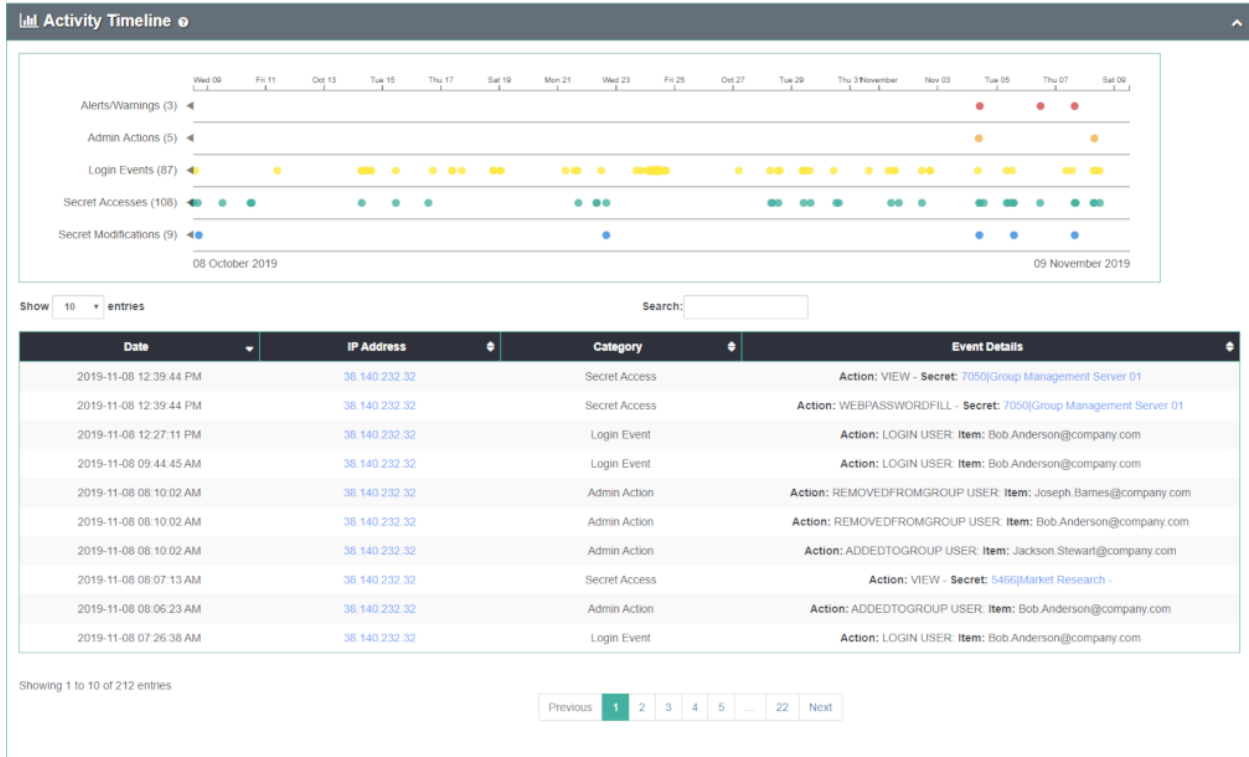
Most Frequent Secrets: an animated representation of the top 20 most accessed Secrets by the User; you can zoom into the graph by scrolling, or right-click on any node or link to view more details

Displaying top 20 links and only data related to entity of interest. Click for User's full Secret Event Graph. (Admin Action Graph.)
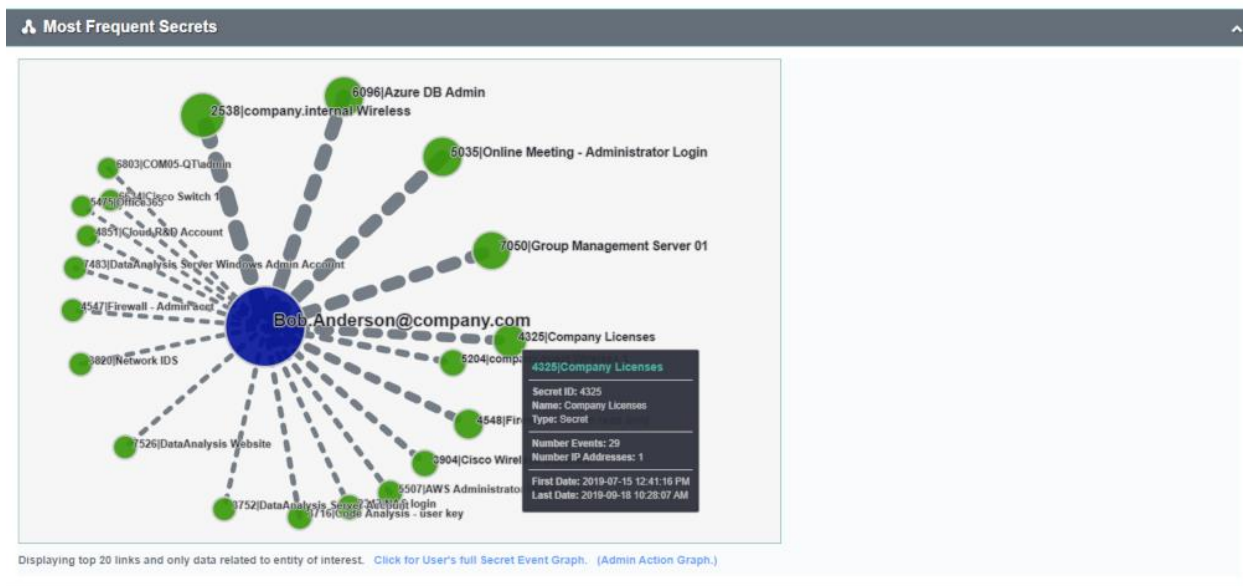
Temporal Behavior: a chart showing all temporal data for the User organized by time of day and day of the week

- the numbers across the bottom indicate the total events involving the User for that time of day

- the values across the right side indicate the number of events involving the User for that day of the week

- the legend at the bottom shows the number of events that correlate to the coloring of the chart blocks

- mouse over a block to get the total number of accesses for that day of week and hour of day

Total number events by hour and day of the week. Click for User's full Secret Event Clock. (Admin Action Clock.)
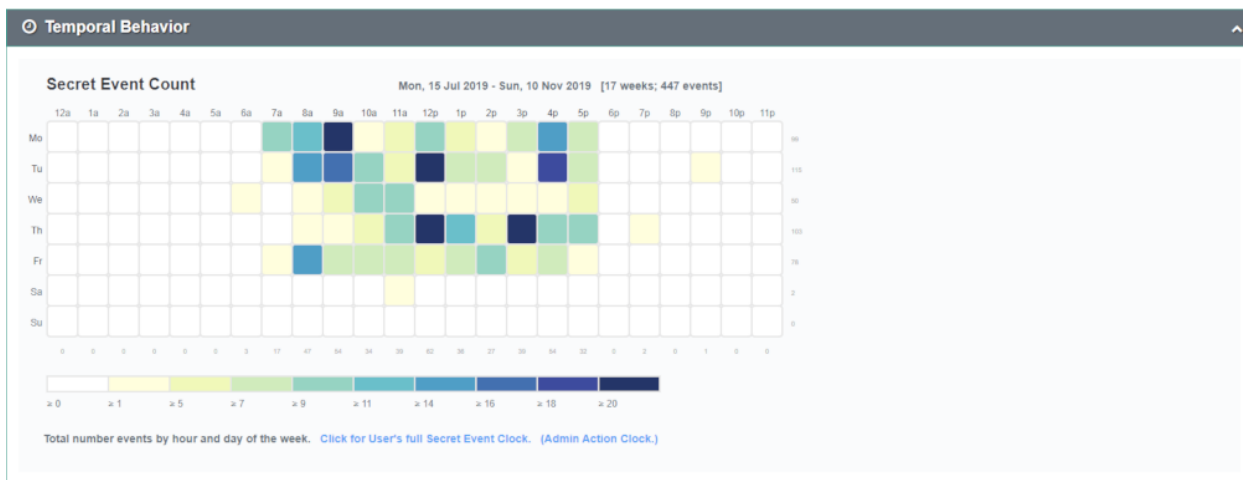
User IP Address History: lists any IP addresses they have accessed Secret Server from

Secret Activity: lists the most recent 500 encrypted Secret accesses, when they occurred, the Secret IDs and names accessed, and how they were accessed

Event Timestamp	Secret ID	Secret Name	Access Method of Secret
2018-03-19 09:19:43 PM	6566	Expense Tracking System	WEBPASSWORDFILL
2018-03-19 09:19:35 PM	6566	Expense Tracking System	VIEW
2018-03-19 08:14:44 PM	6102	CRM - personal account	WEBPASSWORDFILL
2018-03-19 08:14:18 PM	6102	CRM - personal account	WEBPASSWORDFILL
2018-03-19 08:14:11 PM	6102	CRM - personal account	VIEW
2018-03-19 03:45:14 PM	7025	heller.namebiz	WEBPASSWORDFILL
2018-03-19 03:43:47 PM	7025	heller.namebiz	WEBPASSWORDFILL
2018-03-19 03:43:43 PM	5737	Rental Apartment - 1	VIEW
2018-03-19 03:43:43 PM	5737	Rental Apartment - 1	WEBPASSWORDFILL
2018-03-19 03:43:34 PM	7025	heller.namebiz	VIEW

Showing 1 to 10 of 500 entries

Previous 1 2 3 4 5 … 50 Next

Secret Server Administrative Actions: lists any administrative actions the User has performed in Secret Server, when it occurred, what the specific actions was, and if it affected any Secret Server Users

Event Timestamp	Administrative Action	Event Item (Item Type)	Event Details
2018-01-03 10:36:04 AM	ADDEDTOGROUP	Joseph.Barnes@company.com (USER)	
2017-12-28 12:52:50 PM	ADDEDTOGROUP	ADMIN_Hannah.Green@company.com (USER)	
2017-12-28 12:52:50 PM	ADDEDTOGROUP	ADMIN_Jose.Powell@company.com (USER)	
2017-12-11 05:26:52 PM	ADDEDTOGROUP	Bob.Anderson@company.com (USER)	
2017-12-11 05:26:02 PM	REMOVEDFROMGROUP	Bob.Anderson@company.com (USER)	
2017-12-11 05:26:02 PM	REMOVEDFROMGROUP	Bob.Anderson@company.com (USER)	
2017-12-11 05:25:51 PM	ADDEDTOGROUP	Bob.Anderson@company.com (USER)	
2017-12-11 05:25:51 PM	ADDEDTOGROUP	Bob.Anderson@company.com (USER)	
2017-12-02 11:24:11 PM	DISABLE	Jeffrey.Jenkins@company.com (USER)	
2017-11-28 04:32:43 PM	REMOVEDFROMGROUP	Bob.Anderson@company.com (USER)	

Showing 1 to 10 of 13 entries

Previous 1 2 Next

IP ADDRESS DETAILS

The **IP Addresses** page (**Details** > **IP Addresses**) lists all IP addresses, their type (Public or Private), City, Region, Country, the number of Secret accesses plus modifications, the number of unique Secrets accessed, the number of unique users accessing Secrets, the number of

administrator actions performed (including logins), and the first and last time Privileged Behavior Analytics observed the IP address in data.

Click on an IP address to open the IP Details page.

The sections display a variety of detailed information.

Activity Timeline: a chart showing when an IP address was used to perform Secret accesses, modifications, and administrative actions in Secret Server (including login events). Any active or past alerts or warnings are shown as well.

- mouse over a colored circle for details on a particular event

- the chart can be panned left and right by dragging or zoomed by scrolling, which also filter data in the table

Date	Category	Event Details
2019-11-08 05:10:36 PM	Secret Access	User: Charles.Cooper@company.com - Action: PASSWORD_COPIED_TO_CLIPBOARD - Secret: 6178|Office Supplies
2019-11-08 05:10:30 PM	Secret Access	User: Charles.Cooper@company.com - Action: VIEW - Secret: 6178|Office Supplies
2019-11-08 04:42:37 PM	Secret Access	User: Charles.Cooper@company.com - Action: PASSWORD_COPIED_TO_CLIPBOARD - Secret: 4778|Benefits Management
2019-11-08 04:42:35 PM	Secret Access	User: Charles.Cooper@company.com - Action: VIEW - Secret: 4778|Benefits Management
2019-11-08 04:25:43 PM	Secret Access	User: Evelyn.Simmons@company.com - Action: WEBPASSWORDFILL - Secret: 7235|Privilege Manager for Windows 60
2019-11-08 04:25:41 PM	Secret Access	User: Evelyn.Simmons@company.com - Action: VIEW - Secret: 7235|Privilege Manager for Windows 60
2019-11-08 03:56:30 PM	Login Event	User: Employee40 - Action: LOGIN USER: Item: Employee40
2019-11-08 03:56:19 PM	Login Event	User: Employee40 - Action: LOGIN USER: Item: Employee40
2019-11-08 03:56:09 PM	Login Event	User: Employee40 - Action: LOGIN USER: Item: Employee40
2019-11-08 03:56:09 PM	Login Event	User: Employee40 - Action: LOGIN USER: Item: Employee40

Temporal Behavior: a chart showing all temporal data for the IP address organized by time of day and day of the week

- the numbers across the bottom indicate the total events involving the IP address for that time of day

- the values across the right side indicate the number of events involving the IP address for that day of the week

- the legend at the bottom shows the number of events that correlate to the coloring of the chart blocks

- mouse over a block to get the total number of events for that day of week and hour of day
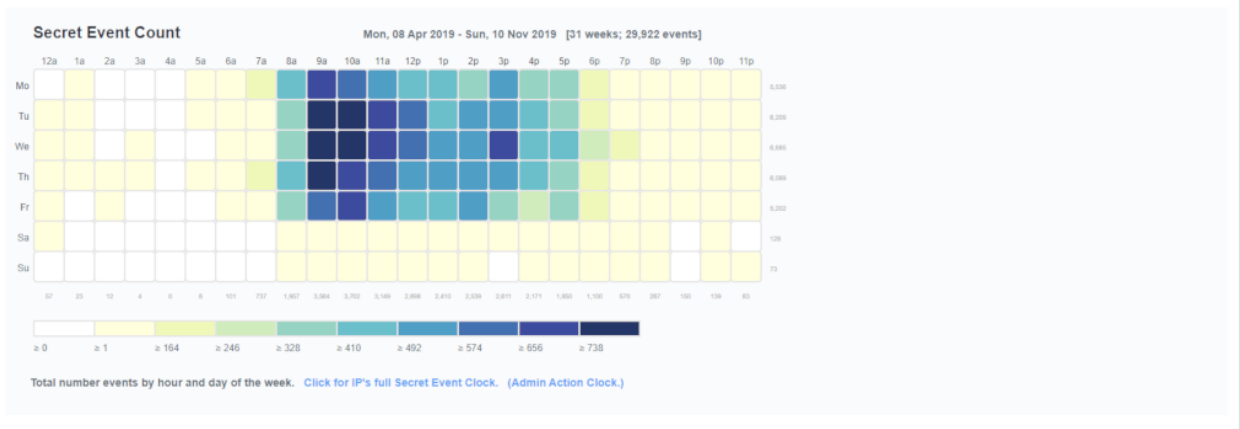
Privileged Behavior Analytics responsive actions

In Privileged Behavior Analytics, anomalous behavior is characterized as an event and assigned a risk level. If this risk level exceeds a pre-defined threshold, the following response actions may be used to triage the potential threat:

- Notification Actions
    - Email Notifications
- Dynamic Actions
    - Access Challenges
    - Webhooks
    - Codehooks

Email Notifications

Each Privileged Behavior Analytics user may set an email and choose to be notified when an event occurs. Upon receipt of an email notification, the anomalous activity should be investigated and remediated if necessary.

Access Challenges

Access Challenges provide automated responses in Secret Server to Privileged Behavior Analytics events. They allow for the dynamic configuration of the trust level in the Secret Access Workflow which is built into Secret Server.

- For example, it may be infeasible to configure Session Recording on every Secret (disk space and CPU limitations) and it may be considered onerous to configure the Approval workflow on every Secret (administrators become desensitized to approving access to every single request).

With Access Challenges, however, these workflows can be enabled dynamically when a user's behavior has become suspicious, helping to ease the tension operationally between security and efficiency.

An expiration duration may be specified for the Challenge, or the Challenge may be valid indefinitely.

- For the **Login** and **Two Factor** Challenges, the challenged user may clear the Challenge by re-authenticating.
- For the other Challenge types, a Secret Server user with Privileged Behavior Analytics permissions must clear the Challenge for the user, or the user must wait for expiration.

For example, a **Requires Approval** Challenge may be configured with a two-hour expiration. This gives the security team a buffer of time to investigate the anomalous activity, while allowing the user to still access their usual Secrets, but in a more restricted workflow.

- **Login**: User must re-authenticate with Secret Server.

- **Two Factor**: User must re-authenticate with Secret Server and the Two Factor Remember Me is expired if set.

- **Require Approval**: User must request approval for accessing any secrets unless they are the only Approver for that secret.

- **Lockout**: User is locked out from Secret Server. This may be configured to include Secret Servers that use SAML and integrated authentication.

- **Session Recording**: User has their sessions recorded for any secrets that are capable of session recording.

  o This session recording is surreptitious, and there is no indicator to the user that they are being recorded.

Webhook

The Webhook action HTTP posts the metadata associated with the anomalous activity event (the user, the time, the actions or secrets accessed) to a user-defined HTTP endpoint.

The Webhook provides the capability to integrate Privileged Behavior Analytics events into many other workflow and security systems. Examples include sending a message to a messaging application such as Slack or creating a case in a ticketing system like ServiceNow.

For more information, see Webhooks in "Privileged Behavior Analytics Administration".

Codehook

Codehooks allow integration with external workflow and security systems where a Webhook is insufficient for the desired behavior. They are user-defined scripts that execute in response to the anomalous activity event.

- Currently Python 2.7 scripts are supported, and Node.js support will be added in a future release.

An example of a Codehook response action is suspending the user's Okta account. This action cannot be achieved by a simple Webhook, despite Okta's REST API, because it requires a two-step process of looking up the user by email, and then using the user's internal Okta identifier to suspend the account.

For more information, see Codehooks in "Privileged Behavior Analytics Administration".

Privileged Behavior Analytics administration

In PBA, most administrative tasks will occur on the **System Settings** page, which is used to set basic configurations for alert notifications and other general settings.

You can navigate to System Settings by clicking on the cogwheel symbol at the top right of any PBA page and choosing System Settings.

Responsive Actions Settings

The **Responsive Actions** section of System Settings is used to configure PBA to take automated action based on user risk score.

Alert Threshold: The numerical value an alert must meet or exceed to send an email and log the event on the Alerts page.

Alert Action: Provides three different automated actions that PBA can take in response to an Alert Event.

- The **Challenge** response can be configured to automatically impose additional controls on a Secret Server user if their actions cause PBA to generate an alert that meets or exceeds the Alert Threshold. The current version of PBA can challenge a user by

    - logging them out of Secret Server

- forcing them to do 2-factor authentication

- locking a user out of Secret Server

- forcing them to request access to any Secrets they access Challenges must be configured on Secret Server as well. More information on how to configure Challenges can be found in Getting Started.

- The **Webhook** response can be configured to integrate with external systems by sending an HTTP post when PBA has a user alert event. Additional information can be found in PBA responsive actions.

- The **Code Hook** response can be configured to integrate with external systems by executing a user provided script when PBA has a user alert event. Additional information can be found in PBA responsive actions.

Warn Threshold: The numerical value a warning needs to meet or exceed to send an email and log the event on the Alerts page.

Warn Action: Provides three different automated actions that PBA can take in response to an Alert Event. See the above *Alert Action* list item for details on automated actions.

Test Actions: Provides an ability to test Responsive Actions to ensure your configuration is correct.
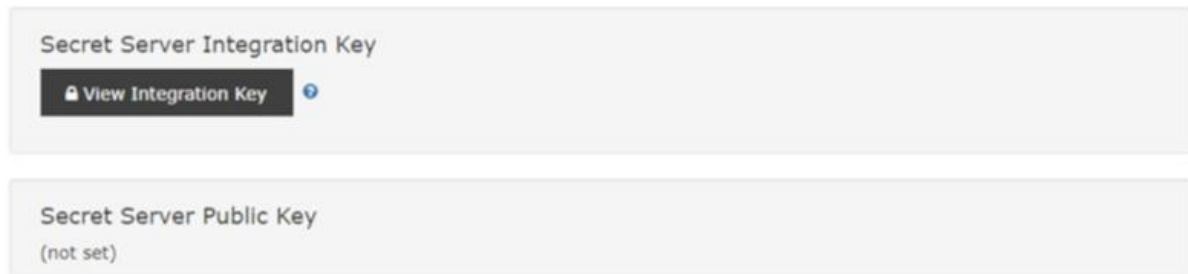
Secret Importance: A page that lists all Secrets and enables changing any of their importance settings for PBA. More important Secrets are more likely to trigger alerts upon User access.

User Watch List: Configuration options to automatically populate the User Watch List with new users and/or users with active alerts and warnings.

Secret Server Integration Settings

The **Secret Server Integration Settings** section is used to configure secure communications between your Secret Server and PBA.

Secret Server Integration Key: A key that provides your Secret Server with credentials and configuration information to upload log data to PBA.

Secret Server Public Key: A one-time RSA public key is entered here to establish communication between Secret Server and PBA.

Time Settings

The **Time Settings** section is used to configure the Timezone and time display format.

Local Timezone: The display of all timestamps can be adjusted to your local time zone. The default time zone is UTC.

Hour Display: 12-hour (AM/PM) or 24-hour (international or "military") time display.

User Settings

The **User Settings** section has password and alert preferences settings.

Account Settings: The link enables changing the password for the account used to access PBA.

Alert Notification Settings: Enables setting the email address for receiving alerts and whether you want to receive alerts or warnings as they occur.

WEBHOOKS

PBA integrates into external workflow and security systems via webhooks. A webhook is a user-defined callback which is executed in response to an event (alert or warning in PBA). Along with Access Challenges and email notifications, webhooks comprise the responsive actions in PBA in response to detection of anomalous activity.

Configuration

URL

This is the URL to which a POST request is sent when an event is created.

Signing Secret

This is optional. If set, a header is set (x-hub-signature) with the SHA256 signature of the body of the email.

Example Signature Verification in Python 2.7:

```
from Crypto.Hash import SHA256

from Flask import requests

\# receive the request

signature = request.headers.get('x-hub-signature',None)

payload = request.data.decode('utf-8')

hmac = SHA256.new(secret)

hmac.update(payload)

expected_signature = "sha256="+hash.hexdigest()

assert signature == expected_signature
```

Encoding

Encoding may be set to either `application/json` or `application/x-www-form-urlencoded`.

However, only `application/json` supports mapped templates (customized POST body; see below).

Custom Headers

This is optional. This field takes JSON formatted headers and adds them to the POST request. For example, to configure basic authentication with username **admin** and password **admin**, you would generate the base64 string of username:password (admin:admin) using the Python example code below, and then set the custom header field to:

```
{"Authorization" : "Basic YWRtaW46YWRtaW4="}

import base64

base64.b64encode("admin:admin")

\# output is: YWRtaW46YWRtaW4='
```

Mapping Template

This is optional. The Mapping Template takes a string as an argument which will be used as the body of the POST. Before doing the POST, PBA will substitute any tokens specified in the mapping template with data from the event. Here are the supported tokens:

- EventId

- UserId

- UserName

- UserEmail

- DisplayName

- StartDate

- EndDate

- RiskScore

- Interval

- Severity

- Threshold

- Hostname

Note that in use, these are enclosed in @ signs, for example: @Severity@

All of the above tokens except Hostname are event fields. The Hostname token denotes the hostname of the PBA instance and may be used to configure a link back to the PBA event from the system which receives the Webhook POST.

If Omitted

If the Mapping Template is not set, then the event (alert or warning) will be serialized and posted as either JSON or urlencoded form data.

- This out-of-the-box formatting of the POST body may be fine for integrating with custom-built REST endpoints, but it is less useful for integrating directly with other products.

- As an example, see the configuration for Creating an Incident with ServiceNow.

Example: Creating a SlackBot

The full instructions for creating a Slack Webhook consumer are available here:

- <https://api.slack.com/incoming-webhooks>

In this example, we forward the PBA events to Slack and they are posted to a channel using a SlackBot in our specified format.

1. Navigate to <https://api.slack.com/apps> create a new app.

2. Turn on **Incoming Webhooks**.

3. Copy your Webhook URL.

4. Click on **Oauth & Permissions** and under **Scope > Select Permission Scopes**, add **Post to a specific channel in Slack** for the channel to which you want the messages posted.

5. In PBA, enable Webhook, and paste the Webhook URL from Step 3 into the **URL** field.

6. Create a Mapping Template with a single JSON field, **text**, and set its value to the format you want the SlackBot to use. For example:

```
{
"text":"PBA Alert: https://@Hostname@/handle_ub_alert/@EventId@\n
Secret Server User: @DisplayName@ (User ID:@UserId@)\n
Time Range: @StartDate@ - @EndDate@\n
Interval: @Interval@\n
Risk Score: @RiskScore@\n
Severity: @Severity@\n
Threshold: @Threshold@"
}
```

Example: Creating an Incident in ServiceNow

The configuration displayed below is an example of Webhook settings that would create an incident in ServiceNow. Here is the Mapping Template:

```
{"assignment_group":"security",
"caller_id":"6816f79cc0a8016401c5a33be04be441",
"description":"Pba Alert\nSecret Server User: @UserName@ (UserId:@UserId@)\n
Pba Event: https://@Hostname@/eventdetails/@EventId@\n
Activity Start: @StartDate@\n
Activity End: @EndDate@\nInterval: @Interval@\n
Risk Score: @RiskScore@\nSeverity: @Severity@\n
```

```
Threshold: @Threshold@", "impact":"1",

"short_description":"PBA Alert on Secret Server User @UserName@. Risk Score: @RiskScore@",

"work_notes":"reported PBA alert"}
```

Notes

Be aware that:

- ServiceNow's REST API uses basic authentication.

- The **caller_id** field should be set to the **id** of the ServiceNow service account used for authentication.

- Thycotic provides a callback link in the incident description: https://@Hostname@/eventdetails/@EventId@ It is also possible to configure an html URL field in ServiceNow and format this as a proper anchor tag.

CODEHOOKS

PBA integrates into external workflow and security systems via **Codehooks**. A Codehook is a user-defined script which is executed in response to an event (alert or warning in PBA).

Along with Access Challenges, Email Notifications and Webhooks, Codehooks comprise the responsive actions in PBA to detection of anomalous activity. A Codehook has the same use case as a Webhook, but is used for an integration which requires more than just an HTTP POST request.

Configuration

The **Script Template** is the script that will be executed as a hook in response to a PBA event.

- It must be written in Python 2.7, and use the pip packages listed here.

- If an unsupported pip package is needed by the script, contact IBM Support to request that it be added.

- Each script invocation has a 30 second timeout.

- Before executing the script, PBA will substitute any tokens specified in the mapping template with data from the event. Here are the supported tokens:

- o EventId

- o UserId

- o UserName

- o UserEmail

- o DisplayName

- o StartDate

- o EndDate

- o RiskScore

- o Interval

- o Severity

- o Threshold

- o Hostname Note that in use, these are enclosed in @ signs, for example: @Severity@

- All of the above tokens except Hostname are event fields. Hostname denotes the hostname of the PBA instance and may be used to configure a link back to the PBA event from the system targeted by the Code Hook.

Example: Suspending an Okta User Account

The script template displayed below is an example of a C script that looks up an Okta user by email (UserEmail token) and suspends the user. This example would be the equivalent of a Lockout Challenge in Secret Server but extrapolated to an external system (Okta). Other potential C actions in Okta would include resetting a user's password or security questions.

```
import urllib2

import urllib

import json

opener = urllib2.build_opener()

user = '@UserEmail@'

opener.addheaders = [('Authorization', 'SSWS <BASE 64 TOKEN>')]

response_str = opener.open('https://<OKTA URL>/api/v1/users?q={0}'.format(user))

response = json.loads(response_str.read())

body = response[0]

user_id = body.get('id',None)
```

```
if not user_id:
 print('user not found')
else:
 #suspend user
 url = 'https://<OKT ULR>/api/v1/users/{0}/lifecycle/suspend'.format(user_id)


 handler = urllib2.HTTPHandler()
 opener = urllib2.build_opener(handler)


 data = urllib.urlencode({})
 request = urllib2.Request(url, data=data)
 request.add_header('Authorization', 'SSWS <BASE 64 TOKEN>')
 request.get_method = lambda: "POST"


 try:
 connection = opener.open(request)
 except urllib2.HTTPError,e:
 connection = e


 if connection.code == 200:
 data = connection.read()
 print('Successfully suspended user: {}'.format(user))
 else:
 print('Failed to suspend user: {}'.format(user))
```

Example: Fax Alerts

PBA supports Fax notifications via codehooks. Codehooks are user-defined scripts which are executed in response to an event (alert or warning in PBA). In this example on configuring Fax Alerts, we use a third-party service, InterFAX, which is an online Fax service with an excellent API.

Configuration

The following script uses the **xhtml2pdf** Python library to convert an HTML document into a PDF for the Fax.

Make sure to replace your InterFAX API credentials and Fax number in the following script and modify the Fax document header in the HTML template with your company and contact information.

```python
from interfax import InterFAX

import shutil

from uuid import uuid4

from xhtml2pdf import pisa

from cStringIO import StringIO

filename = "alert-fax{}.pdf".format(uuid4())

filepath = "/tmp/" + filename


fax_html = """<html><head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=8">

<title>bcl_1363603926.htm</title>

<meta name="generator" content="BCL easyConverter SDK 5.0.08">

<style type="text/css">


body {margin-top: 0px;margin-left: 0px;}


#page_1 {position:relative; overflow: hidden;margin: 102px 0px 174px 120px;padding: 0px;border:
none;width: 696px;}

@page {

 size: a4 portrait;

 @frame header_frame { /* Static Frame */

 -pdf-frame-content: header_content;

 left: 50pt; width: 512pt; top: 50pt; height: 40pt;

 }

 @frame content_frame { /* Content Frame */

 left: 50pt; width: 512pt; top: 90pt; height: 632pt;

 }
```

```
 @frame footer_frame { /* Another static Frame */

 -pdf-frame-content: footer_content;

 left: 50pt; width: 512pt; top: 722pt; height: 150px;

 }

 }


.ft0{font: bold 19px 'Times New Roman';line-height: 22px;}

.ft1{font: bold 14px 'Times New Roman';line-height: 14px;}

.ft2{font: 15px 'Times New Roman';line-height: 17px;}

.ft3{font: bold 14px 'Times New Roman';line-height: 17px;}

.ft4{font: 1px 'Times New Roman';line-height: 1px;}

.ft5{font: bold 16px 'Times New Roman';line-height: 19px;}

.ft6{font: 16px 'Times New Roman';line-height: 19px;}

.ft7{font: 13px 'Times New Roman';line-height: 15px;}


.p0{text-align: left;padding-left: 150px;margin-top: 0px;margin-bottom: 0px;}

.p1{text-align: left;padding-left: 180px;margin-top: 49px;margin-bottom: 0px;}

.p2{text-align: left;margin-top: 0px;margin-bottom: 0px;white-space: nowrap;}


.td0{padding: 0px;margin: 0px;width: 384px;vertical-align: bottom;}

.td1{padding: 0px;margin: 0px;width: 54px;vertical-align: bottom;}


.tr0{height: 26px;}

.tr1{height: 31px;}

.tr2{height: 33px;}


.t0{width: 438px;margin-top: 42px;font: bold 15px 'Times New Roman';}


</style>

</head>

<body>

<div id="header_frame">

<p class="p0 ft0">Privileged Behavior Analytics Alert</p>
```

```
<p class="p1 ft1">{EXAMPLE IT DEPARTMENT}</p>

<table cellpadding="0" cellspacing="0" class="t0">

<tbody><tr>

    <td class="tr0 td0"><p class="p2 ft1">DATE: %s</p></td>

    <td class="tr0 td1"><p class="p2 ft1">TIME: %s</p></td>

</tr><tr>

    <td class="tr1 td0"><p class="p2 ft1">TO: %s</p></td>

    <td class="tr1 td1"><p class="p2 ft1"></p></td>

</tr><tr>

    <td class="tr2 td0"><p class="p2 ft1">FROM: %s</p></td>

    <td class="tr2 td1"><p class="p2 ft1"><span class="ft2"></p></td>

</tr><tr>

    <td class="tr2 td0"><p class="p2 ft1">PHONE: %s</p></td>

    <td class="tr2 td1"><p class="p2 ft3">EMAIL: %s</p></td>

</tr><tr>

    <td class="tr2 td0"><p class="p2 ft1">Number Pages: 1</p></td>

    <td class="tr2 td1"><p class="p2 ft4"> </p></td>

</tr>

</tbody></table>

</div><br>

<div id="content_frame">

<p class="p3 ft5">MESSAGE:</p>

<p class="ft1">Secret Server User: @UserName@ (UserId:@UserId@)</p>

<p class="ft1">Pba Event: https://@Hostname@/eventdetails/@EventId@</p>

<p class="ft1">Activity Start: @StartDate@</p>

<p class="ft1">Activity End: @EndDate@</p>

<p class="ft1">Interval: @Interval@</p>

<p class="ft1">Risk Score: @RiskScore@</p>

<p class="ft1">Severity: @Severity@</p>

<p class="ft1">Threshold: @Threshold@</p>

</span>

</div>

<div id="footer_content">
```

```
<p class="p5 ft5">DISCLAIMER:</p>

<p class="p6 ft7">The information contained in this fax is confidential and property of Example
Inc. Please do not distribute outside the organization.</p>

<p class="p7 ft7">Please check that you have received all pages per the page number above.</p>

</div>

</body></html>"""


to_name = "Security Administrator"

from_name = "Pba Service Account"

from_phone = "+1 669-221-6251 "

from_email = "pbaalerts@example.com"

event_date, event_time = "@StartDate@".split('T')

event_time = event_time[:8]


formatted_fax_html = fax_html % (event_date, event_time, to_name, from_name, from_phone,
from_email)


pdf = StringIO()

pisa.CreatePDF(StringIO(formatted_fax_html.encode('utf-8')), pdf)

resp = pdf.getvalue()


with open (filepath, 'w') as fd:
 pdf.seek (0)
 shutil.copyfileobj (pdf, fd)

interfax = InterFAX(username="<API USERNAME>", password="<API PASSWORD>")

fax = interfax.deliver(fax_number="+99999990", files=[filepath])

fax = fax.reload() # resync with API to get latest status

fax.status # Success if 0. Pending if < 0. Error if > 0
```