PARTNER CERTIFICATION REQUIREMENTS

4 September, 2022

Version 1.8

Table of Contents

Version History

Version	Description	Owner	Date Released
1.0	Initial document creation	Steven Chong	01/13/2022
1.1	Added section 'Monitor for Canceled Screenings'	Steven Chong	01/31/2022
1.2	Updated 'Monitor for Canceled Screenings' with new SMB solution option for displaying result for partially completed reports	Steven Chong	03/15/2022
1.3	Updated webhook/status mapping table for Report Lifecycle support and standardization, created landscape layout and various formatting changes	Ian Boersma	03/22/2022
1.4	Added requirements for listening to account.credentialed and token.deauthorized webhooks.	Ian Boersma	04/07/2022
1.5	Added "Application Data Security" section	Ian Boersma	05/04/2022
1.6	Added info to support customers that have not yet enabled report lifecycle enhancements	Steven Chong	05/27/2022
1.7	Removed requirement that partners should not allow customers to order background checks unless they have Account Hierarchy enabled	Ian Boersma	08/17/2022
1.8	Added additional requirements at request of Support Engineering and Partner teams	Ian Boersma	09/04/2022

Checkr Partner Certification Requirements

Checkr has defined these baseline partner integration certification requirements to ensure that partner integrations will fully support the core background check process for customers, and to ensure consistency among partner integrations. Although partners will have different customer bases and different customer needs, customers all share the same requirements for the core background check process. Partners may implement additional optional Checkr features to enhance their customers' experience after these certification requirements have been met.

A final certification review and approval of your partner integration is required before it may be deployed to Production. This document outlines the components and functionality that must be included in your application for approval to production

A Note on Partners Using Checkr's "Embeds" Components

For those partners implementing Checkr's "Embeds" Javascript library components, the only section of this document that is applicable to you for certification is the "Customer Account Connection" section. Since the Embeds components already have both report initiation and monitoring functionality built in, these solutions should satisfy the certification requirements as-is. You may thus skip review of these additional sections entirely.

Customer Account Connection

OAuth

Checkr provides two ways to connect partner customers to Checkr via OAuth:.

- The Checkr Hosted "Sign-In" flow allows a customer to link an existing Checkr account to your partner application by prompting the admin user to sign into their Checkr account to authorize the connection. ***This flow is always enabled for all partner accounts and cannot be disabled.***

- The Checkr-Hosted "Sign-Up" flow allows a customer to create a Checkr account "on the fly" and connect it to your partner application by gathering the information required directly from your customers. While this flow is enabled via an optional setting on the Partner Application page, Checkr requires that partners support it to provide maximum flexibility for customers.

A sequence diagram of the OAuth process is available here…

REQUIRED: Partners must support the OAuth "sign-up" flow and have this feature enabled in their Partner Application.

Account Credentialing

Once a customer account has been connected to your partner application, you must then listen for the `account.credentialed` webhook from Checkr, which indicates that the account has been approved to run background checks. For OAuth "sign-up" customers that must go through a credentialing process, this webhook could take 1 to 3 days after the OAuth connection process is completed to be triggered; for OAuth "sign-in" customers, this webhook should be sent immediately after the OAuth connection process. Partners must ensure they do not allow the customer to submit any background checks via their platform until they receive this webhook.

REQUIRED: Partners must listen for the "account.credentialed" webhook and only enable background check ordering for their customer once this is received.

Customer Account Disconnection

When a customer decides to discontinue using your partner integration with their Checkr account, they can initiate a disconnection from within their Checkr Dashboard. This action triggers Checkr to send you a `token.deauthorized` webhook that contains the "access token" of the customer who has disconnected their account from your partner application. Once you receive this webhook, you should disable the ability to submit further background checks for that customer in your platform and present a user-friendly message indicating that the customer's Checkr account is no longer connected to your partner application.

REQUIRED: Partners must listen for the "token.deauthorized" webhook and disable further background check orders for the customer within their platform, once received.

Report Initiation

Use Checkr's APIs to initiate background checks from your partner application.

Packages

In addition to the standard packages created for your partner account (Standard Criminal, Standard Criminal with MVR, Pro Criminal and Pro Criminal with MVR) that all customer accounts inherit by default, your customers may also have defined their own "custom" packages, which you need to retrieve for selection. Use `GET /v1/packages` to retrieve all packages for your customer's account. Additionally, you must ensure you support the use of pagination when executing this API call as some customers may have more than 25 packages, which will cause data set truncation.

REQUIRED: Partners must be able to retrieve all packages created on the customer account for selection, both those inherited from the partner account as well as any customer-defined packages.

Account Hierarchy

Checkr's Account Hierarchy feature allows customers to model their Checkr account to match their business structure. Customers use Account Hierarchy to segment users' access to background check reports and data, streamline selection of Packages at the time of order, tailor application of the Positive Adjudication Matrix (PAM) to targeted business areas, and provide more granular detail on their monthly invoice, among other things.

REQUIRED: Partners must support Account Hierarchy on connected customer accounts, which includes:

1. Sending Work Location information in the Candidate and Invitation API calls.
2. Using `GET /v1/packages` to retrieve all packages.
    a. Using pagination to ensure package data is not truncated or API timeouts occur
3. Using `GET /v1/nodes?include=packages` to retrieve all customer Account Hierarchy nodes (if present in the customer account) and any packages associated with those node, as per the following scenarios:
    a. Using pagination to ensure package data is not truncated or API timeouts occur
    b. If the error "***Sorry, your account is not enabled for Hierarchy***" is returned, this indicates the customer account does not support Account Hierarchy, so the partner can simply display the entire list of packages returned in the `GET /v1/packages` call.
    c. If no error is returned, the partner must utilize the following logic:
        i. If the `GET /v1/nodes?include=packages` API call returns no nodes, list all packages from the `GET /v1/packages` call in the drop-down without any node selection filter.
        ii. If the `GET /v1/nodes?include=packages` API call returns nodes, present a node selection filter (cascading hierarchical picker preferred, using the "tier" value as the node label).
            1. If the customer selects a node with one or more assigned packages, list only those assigned packages
            2. if the customer selects a node that doesn't have any assigned packages, list ALL packages from the `GET /v1/packages` call.

Why?

Account Hierarchy provides significant functionality enhancements for Checkr's customers and provides the foundation for all future enhancements.

Create a Candidate

The first step in submitting a background check to Checkr is to create the initial candidate object. This is accomplished by sending a `POST /v1/candidates` call. The "custom_id" attribute in the `/candidates` endpoint should be used to pass a unique identifier (such as your application's applicant ID) to ensure cross referenceable linkage between your partner application and Checkr. The 'success' response to the `POST /v1/candidates` call returns the Checkr "candidate_id" which can then be used as a parameter in the subsequent `POST /v1/invitations` call to create the invitation.

Why?

The Checkr Hosted-Apply Flow helps simplify the process for your customers and their candidates, by automatically gathering the candidate PII required to process the requested screenings. This flow also presents forms and disclosures to the candidate, based on work_location, and gathers candidate consent for the background check, helping your customers remain compliant.

Create Candidate API Call

Checkr recommends that partners pass the following fields in the POST call when creating a candidate.

REQUIRED: Partners must include all fields marked with an asterisk below when making a POST /v1/candidates call.

- First Name*

- Middle Name (if captured)
- Last Name*
- Zip Code
- Phone
- Email*
- Custom ID (unique ID of the consumer in partner application - used as a cross-reference ID)*
- Work location:*
    - country*
    - state (*required if in the US*)
    - city (*highly recommended if in the US*)

REQUIRED: Partners should NOT send the following data when making a POST /v1/candidates call:

- **SSN** - This does not pre-populate on the Checkr invite, so there is no value in sending it
- **Driver License State & Number** - This does not pre-populate on the Checkr invite, so there is no value in sending it
- **DOB** - While this does pre-populate on the Checkr invite, it should not be sent to avoid exposing this key piece of PII to unintended recipients if the candidate's email is entered incorrectly.

A Note on Support for International Background Checks

If the partner will be supporting submission of both domestic and international background checks from their platform, they must ensure they are able to create and use a distinct Candidate object for each scenario. Due to data privacy laws differing between the United States and other countries, the Work Location data sent with the Candidate record determines in which data center (US or EU) the candidate data will be stored. This means that a Candidate record created with a US work location can only be used against an Invitation with a US work location and likewise for a Candidate record created with a non-US work location. If the partner attempts to attach a candidate to an invitation where the work locations (US or non-US) do not match, an error will be returned.

REQUIRED (if supporting International checks): Partners must create and use distinct Candidate records when supporting both domestic and international background checks.

Create an Invitation

Use the 'candidate_id' included in the 'success' response to the `POST /v1/candidates` call as a parameter to create the invitation.

Create Invitation API Call

Checkr requires partners to pass the following parameters in the POST call when creating an invitation.

REQUIRED: Partners must include all fields marked with an asterisk below when making a POST /v1/invitations call.

- Candidate ID*
- Package*
- Node (required if customer has node-enabled hierarchy)
- Work location:*
  - country*
  - state (*required if in the US*)
  - city (*highly recommended if in the US*)

Work Location

Partners must ensure they are sending the work location for the `POST /v1/candidates` and `POST /v1/invitations` calls that most closely matches the actual location where the candidate will be performing their work. Keep in mind that Checkr requires the ISO 2 char abbreviations for the Country and State, rather than the complete names.

Support Creation of Multiple Background Checks for a Candidate

You must provide the ability to order multiple background checks for a single candidate. You may choose to provide your customers the option to manually order a new background check for an existing candidate, or automate the generation of new background checks as part of your application's workflow.

REQUIRED: Partners must support the ability to submit multiple background checks for a single candidate.

Why?

Candidates may apply to multiple positions and there may be different background check requirements for those positions. There are cases when a customer may have ordered the background check with the wrong package, or may have sent the background check to an incorrect email address. In both cases, your customers will be required to order a new background check for their candidate.

Data Validation and Error Handling

To ensure a minimum of issues when submitting background checks from your application, Checkr requires all partners to ensure they perform basic data validation on all fields of data submitted for the `POST /v1/candidates` and `POST /v1/invitations` calls to ensure all data adheres to Checkr's basic requirements. Additionally, in the case any error is returned from the API, Checkr requires all partners to elegantly handle the error and present the details of what went wrong to the customer end-user. A list of the specific errors that could be returned from Checkr's API when performing POST calls is available in the API docs.

REQUIRED: Partners must ensure they working to minimize and handle any errors returned by Checkr's API via:

- Performing basic data validation on all fields of data being sent to Checkr via the `POST /v1/candidates` and `POST /v1/invitations` calls.

- Handling any errors returned from the API and presenting the error detail to the customer end-user.

Support for “Retrying” Failed Background Check Submissions

Checkr requires that partners have a defined mechanism to allow for any background check submissions that fail to be “retried” by the end-user. As an example, if the partner is able to successfully submit the `POST /v1/candidates` call but the `POST /v1/invitations` call fails, there must be a way to retry submitting the invitations call using the original candidate_id returned from the successful candidate creation call, rather than having to create a brand-new candidate record.

REQUIRED: Partner must ensure they have a defined mechanism for allowing failed background check submissions to be resubmitted by the end-user.

Report Monitoring

Use Checkr’s APIs and webhooks to monitor existing background check orders.

Monitor for Canceled Screenings (aka: “Complete Now”/”Report Lifecycle”)

Checkr provides a feature where any reports in “pending” status can be completed and any pending screenings within the report canceled. The common use case for this is when the report is suspended due to an SSN exception or when there is a major delay in completing a screening at the courthouse level and the customer simply wants to complete the report.

The customer can complete a report by clicking the “Complete Now” button in their Checkr Dashboard for the report in question. Checkr supports the notion of both “fully canceled” and “partially completed” reports which are both described below.

NOTE: All customers will be required to adopt this feature in 2022.

Fully Canceled Reports

For a "fully canceled" report, ALL screenings are canceled. This takes place if none of the screenings have completed when the "Complete Now" action is triggered. This will also trigger the `report.canceled` webhook event, which will include screening IDs for each of the included screenings. While a "fully canceled" report is a somewhat rare occurrence, it can happen with screenings that involve manual processing and don't require an SSN trace (such as drug and occupational health screenings) as this adds to the amount of time the screening remains in "pending" status.

For Partners with larger mid-market/strategic customers that segregate recruiting and adjudication tasks (i.e.: where not all users will have access to the Checkr Dashboard), we recommend that partners make a `GET /v1/screenings/:id` call and retrieve the "cancellation_reason_description" to display the canceled status and reason within the partner's application. At this time there is no way to pull the cancellation reason from the report object itself.

Why?

For larger customers that segregate recruiting and adjudication responsibilities, most users will not have access to the Checkr Dashboard. These users need to know why the report has been canceled. Based on the cancellation reason, the customer can define the next steps for the end users.

For partners that support smaller SMB customers, there is no need to display the cancellation reason.

Why?

Smaller SMB customers users will typically perform both recruiting and adjudication responsibilities and will have access to the Checkr Dashboard. These users will be able to access the report details, including the cancellation reason.

Partially Completed Reports

For the "partially completed" scenario, if at least one screening (including the SSN Trace) is completed when the "Complete Now" action is triggered on the report, all remaining non-completed screenings are canceled. In this case, the `report.completed` (`report.updated` for Assess customers) webhook event will still trigger. For all completed reports, Checkr recommends that you retrieve the value from the new "includes_canceled" field that is returned in the `report.completed` (`report.updated` for Assess customers) webhook payload for all completed reports. This is a Boolean field that indicates if the report was canceled mid-stream and thus was completed with one or more canceled screenings. If "includes_canceled" field is not included in the report webhooks, we recommend that you continue with the webhook processing as if "includes_canceled" returned a "false" value. This is to support customers that are not enabled with report lifecycle enhancements.

Checkr recommends two options for displaying the results of the partially completed reports with canceled screening(s) that returns a 'clear' result on the partner platform. For partners that support SMB customers, the partner should display "Complete" or equivalent status with a visual indicator that the report contains canceled screening(s) after receiving the `report.completed` (`report.updated` for Assess customers) event.

For partners that support larger customers that segregate recruiting and adjudication responsibilities, Checkr recommends not displaying the report result for a partially completed report with "clear" result until a review (adjudication) is performed; the report result should only display once the reviewer/adjudicator engages through the Checkr dashboard and the `report.engaged` webhook event is received. An external process must be defined by the customer for dealing with those edge cases where they do not want to move forward based on the canceled screenings.

Below is the link to the different report lifecycle workflows based on the different customer types that partners support. In addition to the two workflows covered in this section for SMB and larger customers, it includes an optional more complex workflow for larger customers.

https://lucid.app/documents/view/7be41ec8-133c-4db2-9dee-d82212912081

Customer Type	Adjudication and Recruiting Responsibilities	Cancelation Reason	Partially Completed 'Clear" report
SMB	Combined	Do not pull and display	Display "Complete" or equivalent with canceled screening indicator
***Mid-Market/Strategic**	Segregated	Pull and display	Do not display report result until "engaged" (i.e.: leave the status as "Pending")

**There can be cases where Mid-Market customer users may have combined recruiting and adjudication responsibilities. If so, follow the SMB row above.*

NOTE: If your customer market is a combination of SMBs and larger customers, we recommend selecting only one option from the above table.

REQUIRED: Partners must support basic "Complete Now"/"Report Lifecycle" functionality, which includes:

1. Listening for the `report.canceled` webhook and displaying the appropriate status in the partner application
2. Retrieving the value of the "includes_canceled" field in the `report.completed` webhook and displaying the appropriate status in the partner application (see "Webhook and Status Mappings" table below)

Webhooks and Status Mappings

Checkr generates webhook notifications for various events that take place during a report's lifecycle.. Subscribe to these webhooks to display information in your partner application to ensure your customers stay informed of the background check report status without having to always log into the Checkr Dashboard.

To meet Checkr's requirements for a partner integration, you must subscribe to the following webhooks. Checkr recommends you display the listed "Partner Status" label in your application for these events.

REQUIRED: Partners must support ALL webhook scenarios contained in the table below.

Webhook	Report Status	Report Result	Report Adjudication	Recommended Partner Label	Comments
`invitation.created`	-	-	-	Invitation Sent	The invite was sent
`invitation.completed`	-	-	-	Pending	The invite was completed by the candidate and the report was created
`invitation.expired`	-	-	-	Invitation Expired	The invite expired after 7 days
`invitation.deleted`	-	-	-	Invitation Canceled	The invite was canceled via the Dashboard or the API
`report.completed`	complete (includes_canceled = false)	clear	*null*	Clear	The report was completed and no charges were found
`report.completed`	complete (includes_canceled = true)	clear	*null*	Complete w Canceled ⚠ [1] Pending [2]	The report was partially completed with canceled screening(s) and no charges were found
`report.completed`	complete (includes_canceled = true)	*null*	*null*	Canceled	The report was automatically completed and all screenings were canceled. An example of this is an SSN exception

Webhook	Report Status	Report Result	Report Adjudication	Recommended Partner Label	Comments
					that resulted in a report suspension lasting 30 days
`report.completed`	complete	consider	*null*	Needs Review [1] Pending [2]	The report was completed and charges were found.
`report.pre_adverse_action`	complete	consider	pre_adverse_action	Pre Adverse Action [1] Pending [2]	The report has been pre-adverse actioned
`report.post_adverse_action`	complete	consider	post_adverse_action	Not Eligible	The report has been automatically post-adverse actioned (normally 7 days after pre-adverse actioning)
`report.engaged`	complete	*<any value>*	engaged	Clear	The report has been engaged
`report.suspended`	suspended	*null*	*null*	Suspended	The report has been suspended due to an exception not being resolved within 7 days or 2 attempts by the candidate
`report.resumed`	pending	*null*	*null*	Pending	The suspended report has been un-suspended (i.e.: “resumed”)
`report.disputed`	dispute	*null*	*null*	Disputed [1] Pending [2]	The report has been disputed by the candidate (only applicable to reports with a “consider” result)

Webhook	Report Status	Report Result	Report Adjudication	Recommended Partner Label	Comments
`report.canceled`	canceled	*null*	*null*	Canceled	All of the screenings in the background check have been canceled (prior to any of them processing)

[1] = For SMB customers that combine recruiting and adjudication responsibilities

[2] = For larger customers that segregate recruiting and adjudication responsibilities

⚠ = “includes canceled screenings” icon

NOTE: Make sure to use the `result` field in the webhook for screening and report outcomes (clear, consider, or null), and the `status` field for the report’s status (pending, complete, suspended, dispute, or canceled

ETA

Checkr’s ETA offering provides an estimate of when all screenings will complete for each background check report. This estimate is based on Checkr’s proprietary machine learning algorithm that uses turnaround times for verifications and record searches for our most common screenings. Partners should display this ETA in their application to provide up-to-date information on when the background check is expected to complete.

> **Why?**
>
> Checkr provides this estimate to help customers provide a better candidate experience, strategically plan start dates and interview pipelines, and further enable transparency into the overall background check process. Checkr ETA provides a date for the estimated completion of a specific report.

Retrieve the ETA

Subscribe to the `report.updated` webhook and listen for ETA-related updates (where the “previous_attributes” field has a value of “estimated_completion_time”: null). The “estimated_completion_time” field will contain the date-time value of the ETA which can then be displayed in the partner application.

REQUIRED: Partners must display the ETA for the background check report in their application.

Access to Report Details

For those customer users of the partner application that have permissions to access background check information within the Checkr Dashboard a link to the background check report in the Checkr dashboard should be made available within the partner’s application. The URI to the report can be constructed from the data returned in the `invitation.completed` webhook as well as all `report.*` webhook events including `report.created` and `report.completed`.

Why?

Checkr recommends that the partner make available a link to the report in the Checkr Dashboard rather than expose the completed report within your application. This will help to ensure that sensitive candidate data is not exposed, creating data privacy risks. It also provides a more seamless experience to end-users.

If your customer has not implemented SSO with Checkr, after clicking the link users will be required to provide proper authentication to access and view the report.

REQUIRED: Partners must display a link to the background check report in Checkr in their application (and restrict display of this link to only those users that have adjudication privileges in Checkr, if at all possible).

Application Data Security

In addition to supporting the features outlined in the sections above, Checkr requires that partners adhere to appropriate protocols and standards to ensure any candidate personally-identifiable information (PII) collected as part of the background check process is kept secure. Some key points of discussion:

- **Integrity**: Do you store sensitive candidate PII (such as SSN or driver license) in your application? If so, which fields?
- **Confidentiality**: Do your logs contain sensitive candidate PII or client credentials? Attach an example log for the following two API calls:
  - POST https://api.checkr.com/oauth/tokens
  - POST https://api.checkr.com/v1/candidates
- **Authentication**:
  - Describe how a user authenticates to access your application. If you do not establish user identity, describe what actions are available to anonymous users.
  - Describe the methods used to encrypt internal storage of secrets (keys, tokens).

Partner integrations require recertification at regular intervals, or after updates to the partner application. Work with your Checkr Partner Manager to determine when recertification will be required for your application.

Optional Features

The following features are optional, and may be added to your implementation.

Feature	Description	Customer Value
Cancel and Resend Invitation	Allow customers to cancel and resend an invitation.	Better control of invitation management.
Additional webhook subscriptions (statuses)	Use Checkr webhooks (verification, etc) to display additional statuses in your partner application.	Greater visibility into the progression of background checks.
Screening level statuses and results	Display statuses and results of each individual screening in a package.	Visibility into each screening's status and result to understand the report's overall status and result.
Package Prices	Display the price for each available package at the time of order	Visibility into the cost for running each type of background check package at the time of order
Report tags	Display customer-specific information.	Ability to tie the partner application's candidate to the Checkr report using additional data points (like the hiring manager's email).
Report order history	Provide an audit log of who ordered what when.	Visibility on who ordered the background check and when for background check tracking and monitoring.
Assess	Enable support for Checkr Assess for your partner integration. Checkr Assess allows customers to create custom rules that are applied to candidate records and	The ability to fine tune their adjudication process to provide more consistent review of returned records.

	provide an assessment value such as Eligible or Review, enabling greater consistency and speed in adjudication. Subscribe to the report.updated webhook event to pull the Assess tag.	
Continuous Checks (C-Crim and C-MVR) is a post-hire solution, available only to employees and volunteers.	Allow your customers to automatically enroll candidates in Checkr's Continuous services based on the completed report's status of Clear, or the adjudication result of Engaged. Checkr's Continuous Check services monitor enrolled workers for any new records that may lead to generation of a standard background check report.	Automated enrollment and unenrollment of employees to Continuous Crim or Continuous MVR without manual involvement.
Subscriptions (available only to employees and volunteers)	Checkr's Subscriptions allow customers to set the interval at which a background check is re-run for an existing employee or volunteer. Most customers choose to re-run background checks annually.	Automated re-runs of background checks, no manual order required.

Auto progression	Trigger the next step in your partner application workflow based on the background check's result.	Automatic progression in the workflow without the need to continuously check for the background check's result.
Account Unlinking	Allow the customer to unlink their account from your partner application.	Ensures customers have an easy way to unlink their account from your partner account.