Crunchtime Kitchen V2 Webhook

Reference Guide

Document Details

Document #: 160_CHIG_260420

Version Release Date: July 2026

Note: This document is intended for technology professionals developing integrations with Crunchtime Kitchen webhooks

Overview

This document provides general Webhook information for Crunchtime Kitchen.

Download the latest version of this document here:
https://links.crunchtime.com/CKRG-UsingCrunchtimeKitchenWebhooks

… or scan the QR code to the left.

Contents

General Webhook Information 4
Test Event Sample 4
Kitchen Server Requirements 5
Security 5
Encrypted Communication 5
Authentication 5
Webhook Requirements 5
Data Format 5
HTTP Method 6
HTTP Custom Header 6
Error Codes and Handling 6
Retry Policies 6
Latency 7
Kitchen Webhook Types and Sub-Types 7
Order Events 7
Item Events 7
Status Events 8
Event Sub-Types (OrderUpdated) 8
Event Sub-Types (ItemUpdated) 9
Payload Field Reference 10
Common Fields (All Hooks) 10
OrderCreated / OrderReady 11
OrderUpdated 12
OrderBumped 14
OrderArrived 15
OrderCancelled 16
ItemCreated 17
ItemUpdated 18
ItemRemoved 19
PartnerUpdate 19
GuestUpdate 20
TimingUpdate 20
VehicleUpdate 20
DestinationUpdate 21

© 2026 Crunchtime - All Rights Reserved | www.crunchtime.com |

StatusUpdate 21
Configuration 22
Example Payloads 22
OrderCreated Example 23
OrderUpdated Example 25
OrderBumped Example 26
OrderArrived Example 27
OrderReady Example 29
OrderCancelled Example 31
ItemCreated Example 32
ItemUpdated Example 33
ItemRemoved Example 34
PartnerUpdate Example 35
GuestUpdate Example 36
TimingUpdate Example 37
VehicleUpdate Example 38
StatusUpdate Example 39
DestinationUpdate Example 40
OrderUpdated and StatusUpdate Hooks 41
Kitchen Status Definitions 43
Date 44
Author 44
Page(s) Changed 44
Changes 44

© 2026 Crunchtime - All Rights Reserved

General Webhook Information

Webhooks are user-defined HTTPS callbacks. Sometimes referred to as “reverse APIs,” a webhook allows you to skip a step by receiving updated data of interest without polling for changes. Whenever new data is available, the Crunchtime Enterprise services will call the HTTPS endpoint on your system to send the data.

> **Note**: Crunchtime can fire on demand a Test Event to confirm the service can successfully post to your endpoint. The Test Event includes the same headers as production webhook events but contains a limited data payload.

The Test Event can be fired for any of the webhook services noted in this document. The “X-QSR-Event” header will contain the event type "TestEvent."

Test Event Sample

Sample Test Header

```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "23",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-33bfd4999f612d67bc0e165caa7f4702-7c1140dc3fd7f434-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "TestEvent",
  "X-Qsr-Hmac": ""
}
```

Sample Test Body

```
This is a test message.
```

© 2026 Crunchtime - All Rights Reserved

Kitchen Server Requirements

For v2 Kitchen webhooks, the minimum compatible version of Crunchtime Kitchen is 2026.2. If your in-store Kitchen Server is an earlier version, you must upgrade to at least 2026.2 to take advantage of v2 webhook features.

The in-store Kitchen Server must be able to reach two new Crunchtime Enterprise endpoints for v2 webhooks to be delivered properly. If your network security setup requires whitelisting individual URLs, please add the following endpoints to your configuration:

https://entcore-webhookv2-api.qsr.cloud

https://entcore-webhookv2-configuration-api.qsr.cloud

A full list of all whitelist URLs for Kitchen Server can be found in the Crunchtime Kitchen Installation Guide: https://links.crunchtime.com/CHIG-InstallCrunchtimeKitchen

Security

All webhook payloads are delivered as structured JSON with optional HMAC-SHA256 authentication, with a signature provided in the header.

Encrypted Communication

Crunchtime Enterprise supports encryption to protect communications between the Crunchtime Enterprise and your HTTP endpoint.

Note: Self-signed certificates are currently not supported.

Crunchtime Enterprise supports the TLS cryptographic protocol. Support for SSLv3 is officially deprecated.

Authentication

Crunchtime highly recommends that you password protect your web services so that only you and Crunchtime can access them. Crunchtime Enterprise supports HTTP Basic Authentication. You may provide an HMAC-SHA256 key for additional security. HMAC keys should contain only ASCII characters and may be up to 200 characters in length.

Webhook Requirements

Data Format

Data is sent and received in JSON format unless otherwise specified. Crunchtime Enterprise will specify “application/json” in the “Accept” header in all requests.

© 2026 Crunchtime - All Rights Reserved

HTTP Method

All webhook requests will be made with HTTP “POST” method unless otherwise specified. Your endpoint must be reachable via public internet.

HTTP Custom Header

Crunchtime Enterprise sends a custom header named 'X-QSR-Event' with a value that indicates the type of event/data change(s) for the published Webhook message.

Error Codes and Handling

Your endpoint must return a 2XX HTTP status code to indicate success (e.g. 200). Any other status code will be interpreted as an error.

Retry Policies

Crunchtime Enterprise will make up to four attempts to send a given webhook to a specific URL endpoint. Crunchtime Enterprise will retry or dispose of a webhook according to the following rules:

- If the endpoint responds with 2XX status code, the webhook delivery is considered successful and no further attempts are made to resend it.
- If the endpoint sends a non-2XX response, the delivery is considered to have failed and the webhook is returned to the queue for reprocessing.
- If the endpoint does not return any response within 10 seconds, the webhook delivery is considered to have timed out and the message is returned to the queue for reprocessing.

If either a failure or timeout scenario occurs, the following rules apply:

- Crunchtime Enterprise will attempt to send the webhook up to three additional times.
- Crunchtime Enterprise will cease sending the webhook if either a successful response is received or after the fourth failed attempt.

Webhook events are only valid in Crunchtime Enterprise for 10 minutes after the event is received. If a webhook expires before 4 retry attempts are made, Enterprise will stop retrying.

Note: *In cases where local Kitchen Server cannot communicate with Crunchtime Enterprise, pending webhook requests are retried for up to one hour, then discarded. Once communication is restored, Crunchtime Enterprise applies its retry policies when attempting to send webhooks to URL endpoints. For longer Enterprise disconnection periods, any pending webhook requests not sent to Crunchtime Enterprise are discarded in the Kitchen Server End of Day process.*

© 2026 Crunchtime - All Rights Reserved

Latency

Under normal circumstances, webhook latency should be negligible and satisfy the needs of a near-real-time service. If you routinely experience webhook latency of several minutes, this could indicate an issue with configuration, network, or webhook services, and you should contact Crunchtime Support.

Kitchen Webhook Types and Sub-Types

There are three major Kitchen webhook categories: Order Events, Item Events, and Status Events. Each category contains multiple hook types, and some of those hook types have multiple sub-hooks.

You may subscribe to one or more hook types, depending on the needs of your application. If you subscribe to a hook type that includes multiple sub-hooks, you will receive all sub-hooks that are applicable to a site's Kitchen configuration.

The general organization of hook types and sub-types is outlined below.

Order Events

Hook Type	Trigger
OrderCreated	A new order or course is created in Crunchtime Kitchen
OrderUpdated	A course's status has changed. Refer to the sub-hook type to see the specific status update (outlined in the Sub-Hook Type table below)
OrderBumped	A course has been bumped from a view
OrderArrived	An order or course has been Arrived in Kitchen
OrderReady	An order is ready for pickup
OrderCancelled	A course or order is voided or removed

Item Events

Hook Type	Trigger
ItemCreated	A new item is added to a course
ItemUpdated	An item's status has changed. Refer to the sub-hook type to see the specific status update (outlined in the Sub-Hook Type table below)
ItemRemoved	An item is voided or removed

© 2026 Crunchtime - All Rights Reserved

Status Events

Hook Type	Trigger
PartnerUpdate	Delivery/order partner information changes
GuestUpdate	Guest information (such as guest name or phone number) changes
TimingUpdate	Pickup/delivery time changes
VehicleUpdate	Vehicle information changes
StatusUpdate	General Kitchen status updates. Status updates that trigger this webhook are: Routed, Bumped, Dumped, UnBumped, Prepared, Delayed, Assembled, PreparedOnce, BumpedOnce, Lagged, Ready
DestinationUpdate	Destination changes

Event Sub-Types (OrderUpdated)

Sub-Hook Type	Description
Completed	Order/course fully completed in the Kitchen
Dumped	Course dumped from display group
Assembled	Course marked as assembled
Prepared	Course marked as prepared
Unprepared	Course unprepared
Cooking	Course started cooking
Uncook	Course uncooked
Void	Course voided
VisitIdUpdate	The VisitUID has been updated
CourseCommentUpdate	A comment on the course has been altered
CourseNameUpdate	The name of a course has been changed
CourseNumberUpdate	A course number has been updated
CourseTypeUpdate	Course type updated
CustomerNameUpdate	The Customer Name field was updated
Destination	The course Destination has been changed
QuoteUpdate	Course Quote updated
TableUpdate	Course Table changed
TerminalUpdate	Course Terminal updated

© 2026 Crunchtime - All Rights Reserved

Note: Once a course has been voided, no additional OrderUpdated webhooks will be sent for that course, except for Completed and Dumped.

Event Sub-Types (ItemUpdated)

Sub-Hook Type	Description
Bumped	Item bumped from a view
Prepared	Item marked as prepared
Unprepared	Item unprepared
Cooking	Item started cooking
Uncook	Item uncooked
Void	Item voided
Delayed	Item is being held in delay due to time delay routing
TimeStatus	Item time status changed
Changed	Generic item change
BumpedFrom	Item bumped from specific view
BumpedTo	Item bumped to specific view

© 2026 Crunchtime - All Rights Reserved

REFERENCE GUIDE

Payload Field Reference

The following sections detail the fields included in each webhook payload. All payloads include the CommonMessageData fields plus hook-specific data.

Wherever possible, payload data has been flattened to remove arrays and data hierarchy.

Note: Empty fields will always be omitted from the payload.

Common Fields (All Hooks)

Field	Type	Description
visitUid	string (GUID)	Unique visit identifier
orderUid	string (GUID)	Unique order identifier
checkNumber	string	Transaction/check number
hookType	string	Primary hook type (e.g., "OrderCreated")
eventType	string	Sub-hook type (e.g., "CourseCreated")
createdTime	string (ISO 8601)	Webhook creation timestamp
hookUid	string (GUID)	Unique webhook identifier
siteUid	string	Site unique identifier
siteId	integer	Site ID
trainingOrder	boolean	Whether this is a training order

OrderCreated / OrderReady

The OrderCreated webhook fires when a new course is created in Crunchtime Kitchen. For future orders or courses that are held in delay, the OrderCreated webhook will be sent when the course is first released to Kitchen.

The OrderReady webhook is sent when a course has reached the threshold for entering Ready status. The definition of "ready" is configurable by individual restaurant concepts.

Includes: Common Fields + Guest + Course + Table + Partner + Timing + Vehicle

Field	Type	Description
guestFirstName	string	Guest first name
guestLastName	string	Guest last name
guestLoyaltyId	string	Loyalty program ID
guestPhoneNumber	string	Contact phone number
guestPhoneNumberType	string	Phone type (mobile/home/work)
guestEmail	string	Email address
guestContactType	string	Guest's preferred contact method. Options are *None* or *SMS*.
courseComment	string	Course-level comment
courseType	string	Course type (DineIn, DriveThru, etc.)
courseName	string	Course name
courseNumber	integer	Course number
customerName	string	Customer name for display
tableName	string	Table identifier
tableSection	string	Table section
zoneId	integer	Zone set ID
zoneDisplay	string	Zone display name
guestCount	integer	Number of guests
deliveryPartnerId	string	Delivery partner order ID
deliveryPartnerName	string	Delivery partner name
orderPartnerId	string	Online ordering partner ID
orderPartnerName	string	Online ordering partner name
destinationId	integer	Destination ID
destinationName	string	Destination name
destinationOrderChannel	string	Order channel (Web, Mobile, etc.)

Field	Type	Description
destinationType	string	Destination type
pickupTime	string (ISO 8601)	Scheduled pickup time
scheduledDeliveryTime	string (ISO 8601)	Scheduled delivery time
vehicleMake	string	Vehicle make
vehicleModel	string	Vehicle model
vehicleColor	string	Vehicle color

OrderUpdated

Includes: Common Fields + Status + Course + Table

This webhook fires as a result of many different status or data update triggers. The full list of possible triggers is outlined in the **OrderUpdated Sub-Hook Types** table above.

For this hook type, Crunchtime guarantees delivery of the following sub-types: CourseCreated → Prepared → Assembled → Completed. Other sub-types will be sent if applicable, but the four listed should be expected for every transaction regardless of Kitchen configuration.

> **Note**: These guaranteed status updates may not arrive in chronological order, but all of them should be delivered for every transaction.

Field	Type	Description
currentStatus	string	Current status (e.g., "Completed", "Prepared")
courseComment	string	Course comment
courseType	string	Course type
courseName	string	Course name
courseNumber	integer	Course number
customerName	string	Customer name
tableName	string	Table name
tableSection	string	Table section
zoneId	integer	Zone ID
zoneDisplay	string	Zone display
guestCount	integer	Guest count
originalValue	string	The original value of the field that was updated
newValue	string	The new value of the field that was updated
isVoided	boolean	Indicates whether a course has been voided. Only appears on Completed and Dumped events for voided courses.

The newValue and originalValue fields are only included in the OrderUpdated webhook payload if the webhook was triggered by a change to another data field (such as destinationID, tableName, or visitUid). OrderUpdated webhooks triggered by status changes will not include these fields.

The isVoided field is only included in the OrderUpdated payload for webhooks with the eventType Completed or Dumped, and only if the Completed or Dumped events are for a course that has been voided.

© 2026 Crunchtime - All Rights Reserved

OrderBumped

The OrderBumped webhook is generated when a course is bumped from a view. The course may still be active on other views in the Kitchen.

Includes: Common Fields + Status + Course + Table + View

Field	Type	Description
currentStatus	string	Current status (e.g., "Completed", "Prepared")
courseComment	string	Course comment
courseType	string	Course type
courseName	string	Course name
courseNumber	integer	Course number
customerName	string	Customer name
tableName	string	Table name
tableSection	string	Table section
zoneId	integer	Zone ID
zoneDisplay	string	Zone display
guestCount	integer	Guest count
sourceViewId	string	Source view ID (Bumped events only)
destinationViewId	string	Destination view ID (Bumped events only)

OrderArrived

The OrderArrived webhook is generated when an order is Arrived.

Includes: Common Fields + Guest + Course + Table + Partner + Timing + Vehicle

Field	Type	Description
courseComment	string	Course comment
courseType	string	Course type
courseName	string	Course name
courseNumber	integer	Course number
customerName	string	Customer name
tableName	string	Table name
tableSection	string	Table section
zoneId	integer	Zone ID
zoneDisplay	string	Zone display
guestCount	integer	Guest count
guestFirstName	string	Guest first name
guestLastName	string	Guest last name
guestLoyaltyId	string	Loyalty program ID
guestPhoneNumber	string	Contact phone number
guestPhoneNumberType	string	Phone type (mobile/home/work)
guestEmail	string	Email address
guestContactType	string	Guest's preferred contact method. Options are *None* or *SMS*.
deliveryPartnerId	string	Delivery partner order ID
deliveryPartnerName	string	Delivery partner name
orderPartnerId	string	Online ordering partner ID
orderPartnerName	string	Online ordering partner name
vehicleMake	string	Vehicle make
vehicleModel	string	Vehicle model
vehicleColor	string	Vehicle color
pickupTime	string (ISO 8601)	Scheduled pickup time
scheduledDeliveryTime	string (ISO 8601)	Scheduled delivery time

© 2026 Crunchtime - All Rights Reserved

OrderCancelled

The OrderCancelled webhook is sent when a course is voided or removed. This occurs when a cancellation message is sent from a POS, the course is removed using KitchenConsole, or the Cancel Order base action is used in Kitchen.

Includes: Common Fields + Cancellation Data

Field	Type	Description
cancelledTime	string (ISO 8601)	Cancellation timestamp
cancelReason	string	Reason: "CheckDeleted", "CourseDeleted", "CourseVoided"

© 2026 Crunchtime - All Rights Reserved

ItemCreated

The ItemCreated webhook fires when a new top-level item is added to a course.

Includes: Common Fields + Item Insert Data + Destination + Common Item Data

Field	Type	Description
destinationId	integer	Destination ID
destinationName	string	Destination name
itemDescription	string	Item description
itemId	integer	Item ID
itemNumber	integer	Internal item number
posItemNumber	integer	POS item number
itemTimeStatus	string	Time status
itemQuantity	integer	Quantity
price	float	Item price
posItemQuantity	integer	POS quantity
itemPlateTime	integer	Plate time
itemModifierId1/2/3	integer	Modifier IDs
itemModifierName1/2/3	string	Modifier names
forecastPreparationTime	integer	Forecast prep time
itemTrait	string	Item trait
itemType	string	Item type
posCourseNumber	integer	POS course number
posParentItemNumber	integer	POS parent item number
parentItemNumber	integer	Parent item number
seatNumber	integer	Seat number

© 2026 Crunchtime - All Rights Reserved

ItemUpdated

The ItemUpdated webhook is generated in response to many different types of changes to an item. The full list of triggers is outlined in the **ItemUpdated Hook Sub-Type** table above.

Please note there are multiple fields that are unique to certain hook sub-types, as charted below.

Includes: Common Fields + Update Item Data + Item Bump Data + Destination + Common Item Data

Update-specific fields vary by event sub-type:

Field	Type	When Included
previousQuantity	integer	Quantity changes
previousPriority	integer	Priority changes
priority	integer	Priority changes
previousDestinationId	integer	Destination changes
currentDestinationId	integer	Destination changes
previousParentNumber	integer	Parent item changes

Bump-specific fields (for Bumped/BumpedFrom/BumpedTo):

Field	Type	Description
sourceViewId	integer	Source view ID
sourceStationId	integer	Source station ID
sourceSosTag	string	Source SOS tag
sourceDisplayGroupId	integer	Source display group
destinationViewId	integer	Destination view ID
destinationStationId	integer	Destination station ID
destinationSosTag	string	Destination SOS tag
destinationDisplayGroupId	integer	Destination display group

Full item data same as ItemCreated.

© 2026 Crunchtime - All Rights Reserved

ItemRemoved

The ItemRemoved webhook is sent when an item in a course is voided or removed. This could be generated by an external void message from a POS or by using the Cancel Item base action within Kitchen.

Includes: Common Fields + Common Item Data

Field	Type	Description
itemDescription	string	Item description
itemId	integer	Item ID
itemNumber	integer	Item number
posItemNumber	integer	POS item number
itemTimeStatus	string	Time status

PartnerUpdate

The PartnerUpdate webhook is generated when any of the Partner data fields (delivery or online partner name or ID) is altered.

Includes: Common Fields + Partner Data

Field	Type	Description
deliveryPartnerId	string	Delivery partner order ID
deliveryPartnerName	string	Delivery partner name
orderPartnerId	string	Order partner ID
orderPartnerName	string	Order partner name
destinationId	integer	Destination ID
destinationName	string	Destination name
destinationOrderChannel	string	Order channel
destinationType	string	Destination type

© 2026 Crunchtime - All Rights Reserved

GuestUpdate

The GuestUpdate webhook is fired when any of the Guest data fields is changed.

Includes: Common Fields + Guest Data

Field	Type	Description
guestFirstName	string	First name
guestLastName	string	Last name
guestLoyaltyId	string	Loyalty ID
guestPhoneNumber	string	Phone number
guestPhoneNumberType	string	Phone type
guestEmail	string	Email
guestContactType	string	Guest's preferred contact method. Options are *None* or *SMS*.

TimingUpdate

The TimingUpdate webhook is generated by a change to the order's scheduled pickup or delivery time.

Includes: Common Fields + Timing Data

Field	Type	Description
pickupTime	string (ISO 8601)	Pickup time
scheduledDeliveryTime	string (ISO 8601)	Delivery time

VehicleUpdate

The VehicleUpdate webhook fires when any of the Vehicle data fields is updated.

Includes: Common Fields + Vehicle Data

Field	Type	Description
vehicleMake	string	Vehicle make
vehicleModel	string	Vehicle model
vehicleColor	string	Vehicle color

DestinationUpdate

The DestinationUpdate webhook is sent when the Destination data on a course is changed.

Includes: Common Fields + Destination Data

Field	Type	Description
destinationId	integer	Destination ID
destinationName	string	Destination name

StatusUpdate

The StatusUpdate webhook is generated by any of several different status change events. The statuses that will trigger this webhook are: Routed, Bumped, Dumped, UnBumped, Prepared, Delayed, Assembled, PreparedOnce, BumpedOnce, Lagged, Ready.

Includes: Common Fields + Status + Quote + Course + Destination + Label + Terminal

Field	Type	Description
currentStatus	string	Current status
quoteTime	integer	Quote time in minutes
maxMinutes	integer	Maximum quote minutes
minMinutes	integer	Minimum quote minutes
courseComment	string	Course comment
courseType	string	Course type
courseName	string	Course name
courseNumber	integer	Course number
customerName	string	Customer name
destinationId	integer	Destination ID
destinationName	string	Destination name
exactQuoteLabel	string	Exact quote label
printQuoteLabel	string	Print quote label
quoteLabel	string	Quote label
pagerId	string	Pager ID
tentNumber	string	Tent/vehicle number
terminal	string	Terminal identifier

© 2026 Crunchtime - All Rights Reserved

Configuration

To enable webhooks, the URL endpoints must be configured in Crunchtime Enterprise. Please contact your Crunchtime Account Representative to provide your URL(s) for receiving Kitchen webhook data. A URL will need to be registered for each site or concept within the company, although the URL for each site and/or concept does not need to be unique. Webhooks will be published to the specified URL when they occur. You may provide Crunchtime with an HMAC-SHA256 key for additional security.

Webhook output to a specific URL can be filtered by an order's Destination ID. Additionally, multiple URL endpoints can be enabled for a given site or concept. This means that orders with different Destination IDs can be directed to separate URL endpoints if, for example, those Destination IDs correspond to different third party ordering partners.

When contacting Crunchtime to enable webhooks, please be prepared to provide the following information:

- The Hook Type(s) you would like to receive
- The URL endpoint(s) for each Hook Type
- Your chosen HMAC key for each endpoint (if desired)
- Destination ID filters (if desired) you would like to apply to any Hook Type/URL combination

The OrderReady Hook Type requires additional configuration on the part of the Crunchtime Kitchen user. Kitchen Portal must be used to specify the conditions that define an order as "ready."

Example Payloads

All Kitchen webhooks will be sent using the HTTPS Method POST.

All hooks will contain an HTTPS header with the custom X-QSR-Event field that specifies the Hook Type.

> **Note**: The examples below contain all possible data fields for the specified webhook. Actual payloads may contain fewer fields if Kitchen does not have data for any of the potential fields. Your application should be capable of handling all potential data fields for a Hook Type for maximum flexibility.

OrderCreated Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "1152",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-5dca14baec7f7c30601fdad6d89f3b9e-4b684b61d66f7f2f-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "OrderCreated",
  "X-Qsr-Hmac": ""
}
```


© 2026 Crunchtime - All Rights Reserved

Body


```
{
		"checkNumber" : "10128",
		"courseComment" : "comment",
		"courseName" : "ENTREES",
		"courseNumber" : 1,
		"courseType" : "Entree",
		"createdTime" : "2026-04-07T18:02:20.915Z",
		"customerName" : "John Smith",
		"deliveryPartnerId" : "11-22-33A",
		"deliveryPartnerName" : "QSRDelivery",
		"destinationId" : 2,
		"destinationName" : "TOGO",
		"destinationOrderChannel" : "OnSite",
		"destinationType" : "DineIn",
		"eventType" : "CourseCreated",
		"guestContactType" : "None",
		"guestCount" : 1,
		"guestEmail" : "Email",
		"guestFirstName" : "John",
		"guestLastName" : "Smith",
		"guestLoyaltyId" : "1000",
		"guestPhoneNumber" : "123-456-7890",
		"guestPhoneNumberType" : "Mobile",
		"hookType" : "OrderCreated",
		"hookUid" : "b1224f04-a191-47c3-be7d-836de54dfd40",
		"orderPartnerId" : "11-22-33A",
		"orderPartnerName" : "QSROnline",
		"orderUid" : "00d0dde8-def6-4547-a6c3-de7fae438fc6",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"tableName" : "101",
		"tableSection" : "Patio",
		"trainingOrder" : false,
		"vehicleColor" : "Red",
		"vehicleMake" : "Make",
		"vehicleModel" : "Model",
		"visitUid" : "00000000-0000-0000-0000-000000000000",
		"zoneDisplay" : "A",
		"zoneId" : 1
}
```


© 2026 Crunchtime - All Rights Reserved | www.crunchtime.com |

OrderUpdated Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "659",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-89a883c0067bcb74aa9941f887287ac0-900edb459866bde6-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "OrderUpdated",
  "X-Qsr-Hmac": ""
}
```


Body (Order Status Update - Cooking)


```
{
		"checkNumber" : "10171",
		"courseComment" : "comments",
		"courseName" : "ENTREES",
		"courseNumber" : 1,
		"courseType" : "Entree",
		"createdTime" : "2026-04-08T14:40:43.823Z",
		"currentStatus" : "Cooking",
		"customerName" : "John Smith",
		"eventType" : "Cooking",
		"guestCount" : 1,
		"hookType" : "OrderUpdated",
		"hookUid" : "b83b0ac0-4d34-4059-8413-b94a3690ebe0",
		"orderUid" : "a56bd1ae-600c-45c4-891f-c3a39411da1f",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"tableName" : "101",
		"tableSection" : "Patio",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000",
		"zoneDisplay" : "A",
		"zoneId" : 1
}
```


© 2026 Crunchtime - All Rights Reserved

Body (Order Data Update - VisitIdUpdate)


```
{
		"checkNumber" : "10172",
		"courseComment" : "comments",
		"courseName" : "ENTREES",
		"courseNumber" : 1,
		"courseType" : "Entree",
		"createdTime" : "2026-04-08T14:48:58.320Z",
		"currentStatus" : "VisitIdUpdate",
		"customerName" : "John Smith",
		"eventType" : "VisitIdUpdate",
		"guestCount" : 1,
		"hookType" : "OrderUpdated",
		"hookUid" : "ded9919e-c966-4889-a6b0-ef6b9f6ae427",
		"newValue" : "f74bbab9-28b0-4cad-9efc-fdcd1bfc332f",
		"orderUid" : "853374aa-6d7f-43e6-96a5-61b41ab885b4",
		"originalValue" : "00000000-0000-0000-0000-000000000000",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"tableName" : "101",
		"tableSection" : "Patio",
		"trainingOrder" : false,
		"visitUid" : "f74bbab9-28b0-4cad-9efc-fdcd1bfc332f",
		"zoneDisplay" : "A",
		"zoneId" : 1
}
```


OrderBumped Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "711",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-63f136538c1bb6747ed0a6de94b36e89-c82c370788fe5d5b-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "OrderBumped",
  "X-Qsr-Hmac": ""
}
```

Body


```
{
		"checkNumber" : "10171",
		"courseComment" : "comments",
		"courseName" : "ENTREES",
		"courseNumber" : 1,
		"courseType" : "Entree",
		"createdTime" : "2026-04-08T14:43:25.250Z",
		"currentStatus" : "Bumped",
		"customerName" : "John Smith",
		"destinationViewId" : "111",
		"eventType" : "Bumped",
		"guestCount" : 1,
		"hookType" : "OrderBumped",
		"hookUid" : "ef731496-4a5e-43ef-aae4-fc98d2bf1001",
		"orderUid" : "a56bd1ae-600c-45c4-891f-c3a39411da1f",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"sourceViewId" : "111",
		"tableName" : "101",
		"tableSection" : "Patio",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000",
		"zoneDisplay" : "A",
		"zoneId" : 1
}
```


OrderArrived Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "1232",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-06f2818f8a2efd16303ca7172599df0d-62856068393cbe1f-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "OrderArrived",
  "X-Qsr-Hmac": ""
}
```


© 2026 Crunchtime - All Rights Reserved

Body


```
{
        "checkNumber" : "10172",
        "courseComment" : "comments",
        "courseName" : "ENTREES",
        "courseNumber" : 1,
        "courseType" : "Entree",
        "createdTime" : "2026-04-08T14:48:58.321Z",
        "customerName" : "John Smith",
        "deliveryPartnerId" : "11-22-33A",
        "deliveryPartnerName" : "QSRDelivery",
        "destinationId" : 2,
        "destinationName" : "TOGO",
        "destinationOrderChannel" : "OnSite",
        "destinationType" : "DineIn",
        "eventType" : "Arrived",
        "guestContactType" : "None",
        "guestCount" : 1,
        "guestEmail" : "Email",
        "guestFirstName" : "John",
        "guestLastName" : "Smith",
        "guestLoyaltyId" : "1000",
        "guestPhoneNumber" : "123-456-7890",
        "guestPhoneNumberType" : "Mobile",
        "hookType" : "OrderArrived",
        "hookUid" : "b8481a2e-ceab-4e6f-818b-350b1df3fabb",
        "orderPartnerId" : "11-22-33A",
        "orderPartnerName" : "QSROnline",
        "orderUid" : "853374aa-6d7f-43e6-96a5-61b41ab885b4",
        "pickupTime" : "2026-04-08T10:40:00-04:00",
        "siteId" : 1170,
        "siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
        "tableName" : "101",
        "tableSection" : "Patio",
        "trainingOrder" : false,
        "vehicleColor" : "Red",
        "vehicleMake" : "Toyota",
        "vehicleModel" : "Sienna",
        "visitUid" : "f74bbab9-28b0-4cad-9efc-fdcd1bfc332f",
        "zoneDisplay" : "A",
        "zoneId" : 1
}
```

OrderReady Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "1300",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-1e442353dded35f523c60dc0a2cded93-707989006f6775f7-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "OrderReady",
  "X-Qsr-Hmac": ""
}
```


© 2026 Crunchtime - All Rights Reserved

Body


```
{
		"checkNumber" : "10172",
		"courseComment" : "comments",
		"courseName" : "ENTREES",
		"courseNumber" : 1,
		"courseType" : "Entree",
		"createdTime" : "2026-04-08T14:52:32.570Z",
		"customerName" : "John Smith",
		"deliveryPartnerId" : "11-22-33A",
		"deliveryPartnerName" : "QSRDelivery",
		"destinationId" : 2,
		"destinationName" : "TOGO",
		"destinationOrderChannel" : "OnSite",
		"destinationType" : "DineIn",
		"eventType" : "OrderReady",
		"guestContactType" : "None",
		"guestCount" : 1,
		"guestEmail" : "Email",
		"guestFirstName" : "John",
		"guestLastName" : "Smith",
		"guestLoyaltyId" : "1000",
		"guestPhoneNumber" : "123-456-7890",
		"guestPhoneNumberType" : "Mobile",
		"hookType" : "OrderReady",
		"hookUid" : "453a244a-0585-4239-a588-a3fdf061c13c",
		"orderPartnerId" : "11-22-33A",
		"orderPartnerName" : "QSROnline",
		"orderUid" : "853374aa-6d7f-43e6-96a5-61b41ab885b4",
		"pagerNumber" : 0,
		"pickupTime" : "2026-04-08T10:40:00-04:00",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"tableName" : "101",
		"tableSection" : "Patio",
		"tentNumber" : 1,
		"trainingOrder" : false,
		"vehicleColor" : "Red",
		"vehicleMake" : "Toyota",
		"vehicleModel" : "Sienna",
		"visitUid" : "f74bbab9-28b0-4cad-9efc-fdcd1bfc332f",
		"zoneDisplay" : "A",
		"zoneId" : 1
}
```


© 2026 Crunchtime - All Rights Reserved

OrderCancelled Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "468",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-79d1222ef9beab98e1a8f0e8e515f52f-e5d1a039235c045f-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "OrderCancelled",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"cancelReason" : "CheckDeleted",
		"cancelledTime" : "2026-04-08T11:04:25.269-04:00",
		"checkNumber" : "10173",
		"createdTime" : "2026-04-08T15:04:25.269Z",
		"eventType" : "",
		"hookType" : "OrderCancelled",
		"hookUid" : "912cf0e0-3271-4d8d-914d-d4b18e8d8271",
		"orderUid" : "62d8d071-8f7b-4c32-bde7-bf4db3738c0e",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```


© 2026 Crunchtime - All Rights Reserved

ItemCreated Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "653",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-017704d76e6a3dbce9cf3c475e9fd2b2-c0660f1d81654eab-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "ItemCreated",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"checkNumber" : "10106",
		"createdTime" : "2026-04-06T18:17:27.894Z",
		"destinationId" : 1,
		"destinationName" : "Dine In",
		"eventType" : "ItemCreated",
		"hookType" : "ItemCreated",
		"hookUid" : "f320c062-ed65-4993-bf6f-96287ac7efda",
		"itemDescription" : "Burger",
		"itemId" : 3001,
		"itemNumber" : 1,
		"itemQuantity" : 1,
		"itemTimeStatus" : "None",
		"itemTrait" : "Normal",
		"itemType" : "FoodItem",
		"orderUid" : "a6498783-ad75-4a7e-b7c3-22ea82881796",
		"posItemNumber" : 1,
		"seatNumber" : 1,
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```


© 2026 Crunchtime - All Rights Reserved

ItemUpdated Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "651",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-c4a443883c7f2ce76095585620cc7d1f-cbe469883b3ef43f-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "ItemUpdated",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"checkNumber" : "10171",
		"createdTime" : "2026-04-08T14:40:43.827Z",
		"destinationId" : 1,
		"destinationName" : "Dine In",
		"eventType" : "Cooking",
		"hookType" : "ItemUpdated",
		"hookUid" : "96b5510a-23a9-4736-994b-70c85755c8d3",
		"itemDescription" : "Burger",
		"itemId" : 3001,
		"itemNumber" : 1,
		"itemQuantity" : 1,
		"itemTimeStatus" : "Normal",
		"itemTrait" : "Normal",
		"itemType" : "FoodItem",
		"orderUid" : "a56bd1ae-600c-45c4-891f-c3a39411da1f",
		"posItemNumber" : 1,
		"seatNumber" : 1,
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```


© 2026 Crunchtime - All Rights Reserved

ItemRemoved Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "503",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-ddc17a74401f648d4b9f3fa77505f856-3f731b601f616c38-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "ItemRemoved",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"checkNumber" : "10173",
		"createdTime" : "2026-04-08T14:55:13.157Z",
		"eventType" : "Void",
		"hookType" : "ItemRemoved",
		"hookUid" : "555033ce-039d-4ef4-8142-c4b004dd12fe",
		"itemDescription" : "Burger",
		"itemId" : 3001,
		"itemNumber" : 1,
		"itemTimeStatus" : "Normal",
		"orderUid" : "62d8d071-8f7b-4c32-bde7-bf4db3738c0e",
		"posItemNumber" : 1,
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```


© 2026 Crunchtime - All Rights Reserved

PartnerUpdate Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "669",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-11b5eee6f313b34acd8261f93a95d10d-95fcee756bdd3dd7-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "PartnerUpdate",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"checkNumber" : "10174",
		"createdTime" : "2026-04-08T15:22:02.591Z",
		"deliveryPartnerId" : "112233A",
		"deliveryPartnerName" : "CrunchtimeDelivery",
		"destinationId" : 2,
		"destinationName" : "TOGO",
		"destinationOrderChannel" : "OnSite",
		"destinationType" : "DineIn",
		"eventType" : "OrderPartner",
		"hookType" : "PartnerUpdate",
		"hookUid" : "8991fbc9-afea-4640-a88e-857553e81731",
		"orderPartnerId" : "11-22-33",
		"orderPartnerName" : "CrunchtimeOnline",
		"orderUid" : "e14ac4ea-d05a-4f26-9783-dc845695b240",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```

GuestUpdate Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "603",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-59a153a7ffbc43d6c599b47ab8209226-f398da017b111b1f-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "GuestUpdate",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"checkNumber" : "10174",
		"createdTime" : "2026-04-08T15:24:48.805Z",
		"eventType" : "GuestUpdate",
		"guestContactType" : "None",
		"guestEmail" : "Email",
		"guestFirstName" : "John",
		"guestLastName" : "Jones",
		"guestLoyaltyId" : "1000",
		"guestPhoneNumber" : "818-456-7890",
		"guestPhoneNumberType" : "Mobile",
		"hookType" : "GuestUpdate",
		"hookUid" : "22c01f20-32e2-43ad-8992-9f71bc20fa75",
		"orderUid" : "e14ac4ea-d05a-4f26-9783-dc845695b240",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```


© 2026 Crunchtime - All Rights Reserved

TimingUpdate Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "435",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-34564a58ec1685b6c77314d74bc35e3e-011d016776edace1-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "TimingUpdate",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"checkNumber" : "10174",
		"createdTime" : "2026-04-08T15:24:48.806Z",
		"eventType" : "PickupTime",
		"hookType" : "TimingUpdate",
		"hookUid" : "f60559c7-fb29-4337-bd57-dd95ca6e412c",
		"orderUid" : "e14ac4ea-d05a-4f26-9783-dc845695b240",
		"pickupTime" : "2026-04-08T11:30:00-04:00",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```


© 2026 Crunchtime - All Rights Reserved

VehicleUpdate Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "473",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-c38fc683ba2f3b3b117ce7b598318617-3f8320ba812d0d27-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "VehicleUpdate",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"checkNumber" : "10174",
		"createdTime" : "2026-04-08T15:24:48.806Z",
		"eventType" : "VehicleUpdate",
		"hookType" : "VehicleUpdate",
		"hookUid" : "14b2954a-6b6b-4641-8022-de9ff7ea7cac",
		"orderUid" : "e14ac4ea-d05a-4f26-9783-dc845695b240",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"trainingOrder" : false,
		"vehicleColor" : "Blue",
		"vehicleMake" : "Honda",
		"vehicleModel" : "Civic",
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```


© 2026 Crunchtime - All Rights Reserved

StatusUpdate Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "832",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-b84a69587062569227ce72b02c50f9e9-23d61cb1e0eda08a-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "StatusUpdate",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"checkNumber" : "10107",
		"courseComment" : "comment",
		"courseName" : "ENTREES",
		"courseNumber" : 1,
		"courseType" : "Entree",
		"createdTime" : "2026-04-06T18:17:39.698Z",
		"currentStatus" : "PreparedOnce",
		"customerName" : "John Smith",
		"destinationId" : 1,
		"destinationName" : "Dine In",
		"eventType" : "CourseStatus",
		"exactQuoteLabel" : "5 minutes",
		"hookType" : "StatusUpdate",
		"hookUid" : "eddd0741-b134-4dc5-b21b-99d5c73fa07f",
		"maxMinutes" : 5,
		"minMinutes" : 1,
		"orderUid" : "a6498783-ad75-4a7e-b7c3-22ea82881796",
		"pagerId" : "0",
		"printQuoteLabel" : "About 5 minutes",
		"quoteLabel" : "1-5 min",
		"quoteTime" : 5,
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"tentNumber" : "1",
		"terminal" : "2",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```


© 2026 Crunchtime - All Rights Reserved

DestinationUpdate Example

Header


```
{
  "Request URI": "/receiver?user=jdoe",
  "Accept": "application/json",
  "Connection": "keep-alive",
  "Content-Length": "385",
  "Content-Type": "application/json; charset=utf-8",
  "Traceparent": "00-9c556f202b7984f6eabebae56294b52b-ede7ce1f298f8ec6-01",
  "X-Forwarded-For": "18.214.223.109",
  "X-Forwarded-Port": "443",
  "X-Forwarded-Proto": "https",
  "X-Qsr-Event": "DestinationUpdate",
  "X-Qsr-Hmac": ""
}
```


Body


```
{
		"checkNumber" : "10174",
		"createdTime" : "2026-04-08T15:29:17.044Z",
		"eventType" : "",
		"hookType" : "DestinationUpdate",
		"hookUid" : "44febf8c-f82c-4224-aeba-d601417c2a15",
		"orderUid" : "e14ac4ea-d05a-4f26-9783-dc845695b240",
		"siteId" : 1170,
		"siteUid" : "8a14f16a-c4ec-4712-8019-4a13835cc87f",
		"trainingOrder" : false,
		"visitUid" : "00000000-0000-0000-0000-000000000000"
}
```


© 2026 Crunchtime - All Rights Reserved

OrderUpdated and StatusUpdate Hooks

The OrderUpdated and StatusUpdate webhooks are similar to the PosCheckUpdated events in earlier versions of Kitchen webhooks. These hook types can be used to monitor an order's status as it moves through the Kitchen, but may require some additional finesse.

OrderUpdated is the more granular of the two, and provides greater insight into an order's journey through the Kitchen. The OrderUpdated webhook will be sent when the order enters any of the following statuses:

- Completed
- Dumped
- Assembled
- Prepared
- Unprepared
- Cooking
- Uncook
- Void

Additionally, OrderUpdated webhooks will fire when updates are made to certain pieces of order data. The non-status-related events are updates to the following data fields:

- Destination
- Terminal
- CustomerName
- Table
- Quote
- CourseComment
- CourseName
- CourseNumber
- CourseType

Webhooks with the hookType "OrderUpdated" will include the triggering event in the eventType field, along with the order's status at the time the event occurred, in the currentStatus field.

If you subscribe to the OrderUpdated webhook, you are guaranteed to receive a minimum of 4 webhooks for every qualifying transaction. OrderUpdated webhooks with the eventType CourseCreated, Prepared, Assembled, and Completed will be sent for all orders.

© 2026 Crunchtime - All Rights Reserved

The StatusUpdate webhook is a higher-level look at an order as it moves through the Kitchen, and is ideal for consumers who do not need to monitor an order's every movement or change. StatusUpdate webhooks will fire when courses enter the following statuses:

- Routed
- Bumped
- Dumped
- UnBumped
- Prepared
- Delayed
- Assembled
- PreparedOnce
- BumpedOnce
- Lagged
- Ready

With the exception of the Routed event, when you receive a webhook with the hookType "StatusUpdate," the eventType will be "CourseStatus," which informs you that the course's status has changed, but does not provide specifics. You will need to examine the currentStatus field in the webhook payload to determine the order's status at present. This could be compared to previous status events for the same order, if the specific action that led to the StatusUpdate webhook is not immediately clear from the currentStatus data.

> **Note**: If the triggering event is for a StatusUpdate webhook is a routing event, the eventType will be "Routed."

© 2026 Crunchtime - All Rights Reserved

Kitchen Status Definitions

The currentStatus field may contain one of the following values:

- **Routed** - At least one Item in the Course has been routed to a view. (StatusUpdate hook)
- **Cooking** - At least one Item contained in the Course has been set to "Cooking" or "Fired."
- **PreparedOnce** -The Course has entered Prepared status for the first time. (StatusUpdate hook)
- **Prepared** - All Items contained in the Course have been bumped off all Prep views.
- **Assembled** - Course has been bumped off all Assembler views (if there is no Assembler, the Course updates to Assembled immediately when the Course goes Prepared).
- **BumpedOnce** - The Course has been bumped from all views for the first time. (StatusUpdate hook)
- **Bumped** - The Course has been bumped from all views. (StatusUpdate hook)
- **Completed** - Course and all Items contained in the Course have been bumped off all views Kitchen-wide, including all Spy or Order Ready views. Course may still be available for recall/remake in the Kitchen.
- **Dumped** - Course and all Items have been bumped from all views and are no longer available to be recalled in the Kitchen.
- **Delayed** - Course has been received but held in delay due to Time Delay Routing or Coursing Manager configuration.
- **Lagged** - At least one item in the Course has entered Lagged status (requires specific configuration in Kitchen Server).
- **Uncook** - The Course has had its Cooking status removed (all items in the course previously marked as Cooking have been reset to not cooking).
- **Unprepared** - The Course is no longer in Prepared status (one or more items in the course has had its Prepared status removed).
- **Unbumped** - The Course was previously bumped from a view but has been recalled.
- **Void** - The Course has been voided or cancelled.
- **Ready** - The Course has met the criteria for Order Ready status. Order Ready status is defined by individual restaurant concepts and is typically based on a Course being Prepared or being Bumped from a specific view.

© 2026 Crunchtime - All Rights Reserved