

Webhooks Reference

ABSTRACT

Webhooks enable organizations to trigger automated operations outside of the SalesTim platform, such as in a custom application, or in an automation tool such as Power Automate or Zapier.

TABLE OF CONTENTS [[toc]]

Managing Webhooks

Organizations can manage webhooks from the SalesTim App UI: 1. Open the **Integration** tab 2. Select the **Webhooks** section.

Due to the fact that data may be exchanged outside of your Microsoft 365 environment, webhooks have to be considered as **highly sensitive**. To protect these operations, the SalesTim platform controls webhooks management through RBAC, making any operation related to webhooks accessible only to users granted with one of the following roles: - **Global admin** (Microsoft 365) role - **Teams service admin** (Microsoft 365) role - **Integration Manager** (SalesTim) role

Anatomy of a Webhook

A webhook is defined by the following properties:

```
{
  "id": "7f105c7d-2dc5-4532-97cd-4e7ae6534c07", // {string} Webhook UUID automatically generated during
  "name": "Example Webhook", // {string} Webhook name - Read/Write
  "description": "This is a new webhook", // {string} Webhook description - Read/Write
  "active": true, // {boolean} Webhook status - Read/Write
  "events": [ // {array} Array of events codes that will trigger the webhook - Read/Write
    "team_provisioning_completed", // {string} Event code - Read/Write
    ...
  ],
  "config": { // {object} Webhook http configuration
    "verb": "post", // {string} Http verb used by the the webhook - ReadOnly as currently only `post` is
    "url": "https://example.com/webhook", // {string} Target URL of the webhook - Read/Write
    "content_type": "json", // {string} Http content-type used by the webhook - ReadOnly as currently o
    "secret": "secretClientValue" // {string} Secret value used to authenticate the webhook emitter by the
  }
}
```

Anatomy of a Request

When triggered, the webhook generates an http POST request to its configured url.

Headers

The following headers are included in the request:

```
{
  "X-SalesTim-Webhook": "", // {string} UUID of the webhook that triggered the request.
  "X-SalesTim-Event": "", // {string} Code of the event that triggered the request.
  "X-SalesTim-Delivery": "", // {string} An automatically generated UUID to identify the request.
  "X-SalesTim-Signature": "" // {string} This header is sent if the webhook is configured with a secret
}
```

Payload

A webhook payload contains at least the following properties:

```
{
  "@odata.context": "https://developers.salestim.com/api/webhooks", // {string} Link to the webhook onl
  "tenant": {
    "id": "" // {string} The tenant ID from where the event originates
  }
}
```

User-Agent

The User-Agent for the requests will have the prefix SalesTim-Webhook/ and include the SalesTim current version number.

For instance SalesTim-Webhook/2.1.193

Supported Events

Here are sample payloads for each supported event.

Team Created

- Code: team_created
- Category: team
- Description: Triggered when a team is created, whatever the creation origin (manual, api, using a template...).
- Status:

Sample:

```
{
  "tenant": {
    "id": "79f874ba-8775-44eb-b4d4-b32d9b65cce8"
  },
  "team": {
    "id": "3d8a075c-1226-4c1e-969b-b6d5d65700b1"
  }
}
```

Schema:

```
{
  "type": "object",
  "properties": {
    "tenant": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        }
      }
    },
    "team": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        }
      }
    }
  }
}
```

```

    }
  }
}
}

```

Team Provisioning Completed

- Code: team_provisioning_completed
- Category: team
- Description: Triggered when a team provisioning request based on a template is complete (whether successfully or not).
- Status:

Sample:

```

{
  "tenant": {
    "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",
    "initialDomainName": "contoso.onmicrosoft.com",
    "defaultDomainName": "contoso.com"
  },
  "team": {
    "id": "94735ea1-c097-4cbb-963b-f59bb586fba1",
    "displayName": "Project Team"
  },
  "requester": {
    "id": "f16c50fe-dcb7-4adc-a5ba-2f59f8f50da2",
    "displayName": "John Doe"
  },
  "template": {
    "id": "d8854819-c06f-4680-82f7-cc6d5b230c27",
    "name": "Project Template"
  },
  "metadata": {}
}

```

Schema:

```

{
  "type": "object",
  "properties": {
    "tenant": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        },
        "initialDomainName": {
          "type": "string"
        },
        "defaultDomainName": {
          "type": "string"
        }
      }
    }
  },
}

```

```

    "team": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        },
        "displayName": {
          "type": "string"
        }
      }
    },
    "requester": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        },
        "displayName": {
          "type": "string"
        }
      }
    },
    "template": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        },
        "name": {
          "type": "string"
        }
      }
    },
    "metadata": {
      "type": "object"
    }
  }
}

```

Team Creation Approval Requested

- Code: team_creation_approval_requested
- Category: approval
- Description: Triggered when a user is requesting the creation of a new team using a template that enforces team creation approval.
- Status:

Sample:

```

{
  "tenant": {
    "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a"
  },
  "requester": {
    "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",

```



```

    }
  },
  "template": {
    "type": "object",
    "properties": {
      "id": {
        "type": "string"
      },
      "name": {
        "type": "string"
      },
      "approvers": {
        "type": "object",
        "properties": {}
      },
      "permanentMembership": {
        "type": "object",
        "properties": {}
      }
    }
  },
  "request": {
    "type": "object",
    "properties": {
      "team": {
        "type": "object",
        "properties": {
          "name": {
            "type": "string"
          },
          "description": {
            "type": "string"
          },
          "welcomeMessage": {
            "type": "string"
          }
        }
      },
      "membership": {
        "type": "object",
        "properties": {}
      }
    }
  }
}

```

Team Creation Approved

- Code: team_creation_approved
- Category: approval
- Description: Triggered when a team creation approval is approved.
- Status:

Sample:

```

{
  "tenant": {
    "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a"
  },
  "requester": {
    "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",
    "displayName": "Alice Hawking",
    "mail": "alice.hawking@contoso.com"
  },
  "approval": {
    "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a-5cbf4991-76ef-4a5d-94be-59d72d6e2d7a-5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",
    "approver": {
      "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",
      "message": "Approved!"
    }
  },
  "template": {
    "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",
    "name": "Project Template",
    "approvers": {},
    "permanentMembership": {}
  },
  "request": {
    "team": {
      "name": "Construction Project",
      "description": "Project team for the P100 project",
      "welcomeMessage": "Welcome to the P100 project team!"
    },
    "membership": {}
  },
  "provisioning": {
    "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a"
  }
}

```

Schema:

```

{
  "type": "object",
  "properties": {
    "tenant": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        }
      }
    },
    "requester": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        },
        "displayName": {

```

```

        "type": "string"
    },
    "mail": {
        "type": "string"
    }
},
"approval": {
    "type": "object",
    "properties": {
        "id": {
            "type": "string"
        },
        "approver": {
            "type": "object",
            "properties": {
                "id": {
                    "type": "string"
                },
                "message": {
                    "type": "string"
                }
            }
        }
    }
},
"template": {
    "type": "object",
    "properties": {
        "id": {
            "type": "string"
        },
        "name": {
            "type": "string"
        },
        "approvers": {
            "type": "object",
            "properties": {}
        },
        "permanentMembership": {
            "type": "object",
            "properties": {}
        }
    }
},
"request": {
    "type": "object",
    "properties": {
        "team": {
            "type": "object",
            "properties": {
                "name": {
                    "type": "string"
                }
            }
        },

```



```

        "description": {
            "type": "string"
        },
        "welcomeMessage": {
            "type": "string"
        }
    }
},
"membership": {
    "type": "object",
    "properties": {}
}
},
"provisioning": {
    "type": "object",
    "properties": {
        "id": {
            "type": "string"
        }
    }
}
}
}
}

```

Team Creation Rejected

- Code: team_creation_rejected
- Category: approval
- Description: Triggered when a team creation approval is rejected.
- Status:

Sample:

```

{
    "tenant": {
        "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a"
    },
    "requester": {
        "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",
        "displayName": "Alice Hawking",
        "mail": "alice.hawking@contoso.com"
    },
    "approval": {
        "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a-5cbf4991-76ef-4a5d-94be-59d72d6e2d7a-5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",
        "approver": {
            "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",
            "message": "Rejected!"
        }
    },
    "template": {
        "id": "5cbf4991-76ef-4a5d-94be-59d72d6e2d7a",
        "name": "Project Template",
        "approvers": {},
        "permanentMembership": {}
    }
}

```

```

},
"request": {
  "team": {
    "name": "Construction Project",
    "description": "Project team for the P100 project",
    "welcomeMessage": "Welcome to the P100 project team!"
  },
  "membership": {}
}
}

```

Schema:

```

{
  "type": "object",
  "properties": {
    "tenant": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        }
      }
    },
    "requester": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        },
        "displayName": {
          "type": "string"
        },
        "mail": {
          "type": "string"
        }
      }
    },
    "approval": {
      "type": "object",
      "properties": {
        "id": {
          "type": "string"
        },
        "approver": {
          "type": "object",
          "properties": {
            "id": {
              "type": "string"
            },
            "message": {
              "type": "string"
            }
          }
        }
      }
    }
  }
}

```

```

    }
  },
  "template": {
    "type": "object",
    "properties": {
      "id": {
        "type": "string"
      },
      "name": {
        "type": "string"
      },
      "approvers": {
        "type": "object",
        "properties": {}
      },
      "permanentMembership": {
        "type": "object",
        "properties": {}
      }
    }
  },
  "request": {
    "type": "object",
    "properties": {
      "team": {
        "type": "object",
        "properties": {
          "name": {
            "type": "string"
          }
        },
        "description": {
          "type": "string"
        },
        "welcomeMessage": {
          "type": "string"
        }
      }
    }
  },
  "membership": {
    "type": "object",
    "properties": {}
  }
}
}
}
}
}

```

Endpoints Requirements

Security

Your endpoint must be an HTTPS URL with a valid SSL certificate that can correctly process event notifications.

Responses

The expected success response codes from the target endpoint are 200, 201, 202. If any other code is received, our webhook engine will retry the request following this retry policy:

```
MAX_RETRY = 2 // Maximum number of retry before flagging the delivery as `failed`
RETRY_INTERVAL = 10000 // Number of milliseconds between each retry
```

Verifying Webhooks

Webhooks sent by SalesTim can be verified by calculating a digital signature. If a secret has been defined, the webhook requests will include a **X-SalesTim-Signature** header.

To verify that the request came from SalesTim, compute the HMAC hex digest of the request body, generated using the SHA-256 hash function and the secret as the HMAC key. If they match, then you can be sure that the webhook was sent from SalesTim.

Here are a comprehensive list of examples for multiple languages:

Node

```
const crypto = require('crypto');
const hmac = crypto.createHmac('sha256', 'secret');
hmac.update('Message');
console.log(hmac.digest('hex'));
```

PHP

```
<?php
echo hash_hmac('sha256', 'Message', 'secret');
?>
```

Java

```
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;

public class Test {
    public static void main(String[] args) {
        try {
            String secret = "secret";
            String message = "Message";

            Mac sha256_HMAC = Mac.getInstance("HmacSHA256");
            SecretKeySpec secret_key = new SecretKeySpec(secret.getBytes(), "HmacSHA256");
            sha256_HMAC.init(secret_key);

            byte[] hash = (sha256_HMAC.doFinal(message.getBytes()));
            StringBuffer result = new StringBuffer();
            for (byte b : hash) {
                result.append(String.format("%02x", b));
            }
            System.out.println(result.toString());
        }
        catch (Exception e){
            System.out.println("Error");
        }
    }
}
```

```

    }
}

```

C

```

using System.Security.Cryptography;

namespace Test
{
    public class MyHmac
    {
        public static void Main(string[] args){
            var hmac = new MyHmac ();
            System.Console.WriteLine(hmac.CreateToken ("Message", "secret"));
        }
        private string CreateToken(string message, string secret)
        {
            secret = secret ?? "";
            var encoding = new System.Text.ASCIIEncoding();
            byte[] keyByte = encoding.GetBytes(secret);
            byte[] messageBytes = encoding.GetBytes(message);
            using (var hmacsha256 = new HMACSHA256(keyByte))
            {
                byte[] hashmessage = hmacsha256.ComputeHash(messageBytes);

                var sb = new System.Text.StringBuilder();
                for (var i = 0; i <= hashmessage.Length - 1; i++)
                {
                    sb.Append(hashmessage[i].ToString("x2"));
                }
                return sb.ToString();
            }
        }
    }
}

```

Go

```

package main

import (
    "crypto/hmac"
    "crypto/sha256"
    "encoding/hex"
    "fmt"
)

func ComputeHmac256(message string, secret string) string {
    key := []byte(secret)
    h := hmac.New(sha256.New, key)
    h.Write([]byte(message))
    return hex.EncodeToString(h.Sum(nil))
}

func main() {

```

```
    fmt.Println(ComputeHmac256("Message", "secret"))
}
```

Ruby

```
require 'openssl'
OpenSSL::HMAC.hexdigest('sha256', "secret", "Message")
```

Python (3.x)

```
import hashlib
import hmac

KEY = "secret"
KEY_BYTES=KEY.encode('ascii')
MESSAGE = "Message"
MESSAGE_BYTES=MESSAGE.encode('ascii')
result = hmac.new(KEY_BYTES, MESSAGE_BYTES, hashlib.sha256).hexdigest()

print (result)
```