

# SSO モードの Finesse とサードパーティ小道具を統合

## 目次

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[SSO モードのための相互対話の基本的なモデルの説明](#)

[SSO および NONSSO モードのための gadgets.io.makerequest の設定](#)

## 概要

システムが単一サインオン (SSO) モードにある間、Finesse のサードパーティ小道具の統合のために必要であるものをこの資料に記述されています。例はまた NON SSO モードのために与えられます。

## 前提条件

### 要件

次の項目に関する知識が推奨されます。

- Cisco Finesse
- SSO
- Finesse サードパーティ小道具

### 使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づくものです。

- Cisco Finesse バージョン 11.6
- SSO
- サードパーティ小道具
- サードパーティ他サービス。

本書の情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。このドキュメントで使用するすべてのデバイスは、初期 (デフォルト) 設定の状態から起動しています。対象のネットワークが実稼働中である場合には、どのような作業についても、その潜在的な影響について確実に理解しておく必要があります。

## 背景説明

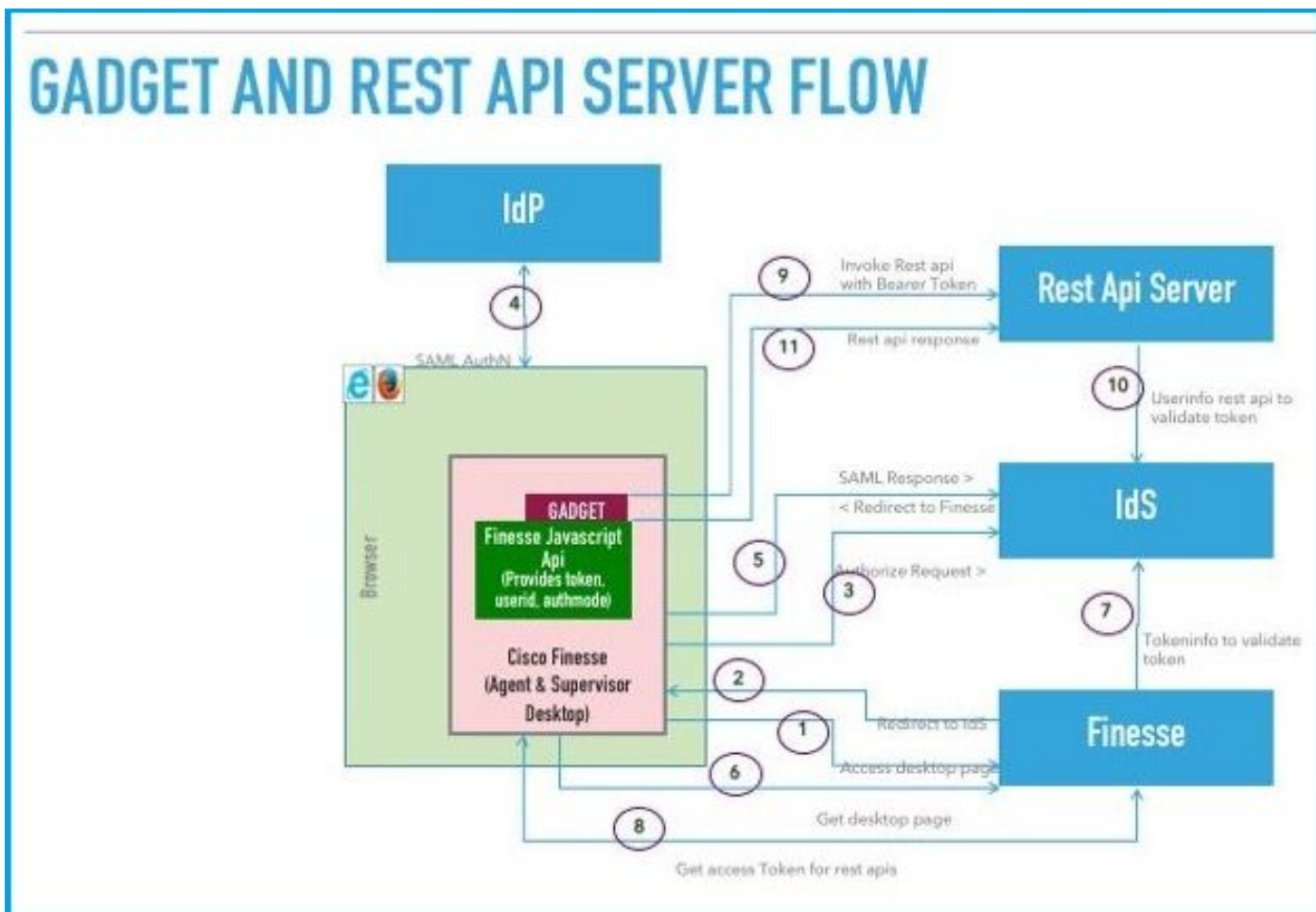
これらはエージェントが SSO または NONSSO のログインおよび認証するに試みる間、第一歩です。

第2ステップは SSO および NONSSO の場合には認証の成功の後で考慮されるどんな必要記述します。

1. デスクトップ ログインの時に、Finesse はシステム Auth モード ( SSO/NONSSO ) を検出する、Auth モードに基づいて、適切な Loginpage は表示する。ユーザは SSO モードの場合には IDP ログイン ページおよび NONSSO モードの場合には Finesse ログイン ページを見ます。
2. 認証の成功の後で、すべての要求はシステム Auth モードに基づいて認証されます。SSO 配備に関しては、うまく解決するすべての要求は要求ヘッダーの一部としてアクセストークンを運びます。トークンは認証の成功のための IDP サーバに対して検証されます。ただし、サードパーティ Web サービスへの要求のために、Auth ヘッダはサードパーティ Web サービスによって設定される認証機構に基づいていました設定されなければなりません。NONSSO 配備の場合には、すべての要求は base64 によって符号化されるユーザ名 およびパスワードの Auth 基本的なヘッダを運びます。All 要求は Finesse ローカルデータベースに対してこの場合検証されます。

## SSO モードのための相互対話の基本的なモデルの説明

このイメージはシステムが SSO モードにあるとき、サードパーティ小道具、Finesse、IDS およびサードパーティ他サービス間の相互対話の基本的なモデルを示します。



画像

イメージで説明される各ステップのための説明はここにあります。

1. エージェント/スーパバイザは Finesse デスクトップ URL をアクセスしました。(例 : <https://finesse.com:8445/desktop>)
2. 認証モードが SSO で、IDS にブラウザをリダイレクトすること検出をうまく解決して下さい。
3. ブラウザは IDS にリダイレクトを承認します要求を送信します。この時点で、IDS はユーザは有効なアクセストークンがあるかどうか検出する。ユーザに有効なアクセストークンがない場合、識別プロバイダ (IdP) への IDS リダイレクト。
4. 要求が IdP にリダイレクトされる場合、IdP はユーザを認証するためにログイン ページを提供します。
5. IdP から SAML アサーションは Finesse デスクトップに戻ってリダイレクトする IDS に送信されます。
6. ブラウザは Finesse デスクトップ ページの GET をします。
7. gets を SAML auth コードの IDS からのアクセストークンうまく解決して下さい。
8. デスクトップはアクセストークンをそれに続く他 API を認証するのに使用されるように得ます。
9. サードパーティ小道具はデスクトップにロードし、auth ヘッダのアクセストークン (所有者) が付いているサードパーティ他 API を呼び出します。
10. サードパーティ他サービスは IDS のトークンを検証します。
11. サードパーティ他応答は小道具に返されます。

## SSO および NONSSO モードのための gadgets.io.makerequest の設定

ステップ 1 : 大騒ぎによってなされる Finesse 他 API コールに関しては小道具は gadgets.io.makeRequest ヘッダの「所有者」許可ヘッダを付加する必要があります。

ステップ 2.小道具は要求パラメーターの中で許可ヘッダ設定されなければなりませんすべての他要求のためのネイティブ gadgets.io.makeRequest コールをする必要があります。

NON SSO 配備に関しては、これは Auth ヘッダです。

```
"Basic " + base64.encode(username : password)
```

SSO 配備に関しては、これは Auth ヘッダです。

```
"Bearer " + access_token
```

アクセストークンは `finesse.gadget.Config` オブジェクトから取得することができます。

```
access_token = finesse.gadget.Config.authToken
```

新しい許可ヘッダ `mustl` は要求パラメーターに追加されます。

```
params[gadgets.io.RequestParameters.HEADERS].Authorization = "Basic " + base64.encode(username : password);
```

```
params[gadgets.io.RequestParameters.HEADERS].Authorization = "Bearer " + access_token;
```

ステップ 3 `getAuthHeaderString` ユーティリティ メソッドはずっと追加された内部ユーティリティです。ユーティリティ。このユーティリティ メソッドは引数として構成オブジェクトを奪取し、許可 ヘッダ スtring を戻します。小道具は要求パラメーターの許可 ヘッダを設定するためにこのユーティリティ メソッドを利用できます。

```
params[gadgets.io.RequestParameters.HEADERS].Authorization=  
finesse.utilities.Utilities.getAuthHeaderString(finesse.gadget.config);
```

**注:** サードパーティ Web サービスへの API 要求に関しては、auth ヘッダはサードパーティ Web サービスによって設定される認証機構に基づいていました設定されなければなりません。小道具開発者に基本的な auth または所有者トークンベースの認証を使用する自由または選択の他のどの認証機構もあります。