Artigo
Vinicius Maranh... · Out. 7, 2020 4min de leitura

Protegendo suas APIs com OAuth 2.0 no InterSystems API Management - Parte 1

Introdução

Hoje em dia existem muitas aplicações que estão usando o Open Authorization framework (OAuth) para acessar recursos de todos os tipos de serviços de maneira segura, confiável e eficiente. O InterSystems IRIS já é compatível com a estrutura OAuth 2.0, na verdade, há um ótimo artigo na comunidade sobre OAuth 2.0 e InterSystems IRIS no seguinte link aqui.

No entanto, com o advento das ferramentas de gerenciamento de API, algumas organizações estão usando-as como um único ponto de autenticação, evitando que solicitações não autorizadas cheguem aos serviços de downstream e desacoplando as complexidades de autorização/autenticação do próprio serviço.

Como você deve saber, a InterSystems lançou sua ferramenta de gerenciamento de API, chamada InterSystems API Management (IAM), que está disponível com a licença IRIS Enterprise (mas não com o IRIS Community Edition). Aqui esta outra ótima postagem na comunidade apresentando o InterSystems API Management.

Esta é a primeira parte de uma série de artigos de três partes que mostram como você pode usar o IAM para simplesmente adicionar segurança, de acordo com os padrões do OAuth 2.0, a um serviço não autenticado implantado no IRIS anteriormente.

Nesta primeira parte, serão fornecidos alguns antecedentes do OAuth 2.0 juntamente com algumas definições e configurações iniciais do IRIS e IAM para facilitar a compreensão de todo o processo de proteção de seus serviços.

Após a primeira parte, esta série de artigos abordará dois cenários possíveis para proteger seus serviços com IAM. No primeiro cenário, o IAM validará apenas o token de acesso presente na solicitação de entrada e encaminhará a solicitação para o backend se a validação for bem-sucedida. No segundo cenário, o IAM irá gerar um token de acesso (atuando como um servidor de autorização) e validá-lo.

Portanto, a segunda parte irá discutir e mostrar em detalhes as etapas necessárias para configurar o cenário 1 e a terceira parte irá discutir e demonstrar as configurações do cenário 2, juntamente com algumas considerações finais.

Se você quiser testar o IAM, entre em contato com seu representante de vendas da InterSystems.

OAuth 2.0 - A Experiência

Cada fluxo de autorização OAuth 2.0 consiste basicamente em 4 partes:

1. Usuário
2. Cliente
3. Servidor de Autorização
4. Proprietário do Recurso

Para simplificar, este artigo usará o fluxo OAuth " Credenciais da Senha do Proprietário do Recurso " , mas você pode usar qualquer fluxo OAuth no IAM. Além disso, este artigo não especificará nenhum escopo.
Nota: Você só deve usar o fluxo de Credenciais da Senha do Proprietário do Recurso quando a aplicação cliente for altamente confiável, pois ela lida diretamente com as credenciais do usuário. Na maioria dos casos, o cliente deve ser uma aplicação primária.

Normalmente, o fluxo de Credenciais da Senha do Proprietário do Recurso segue estas etapas:

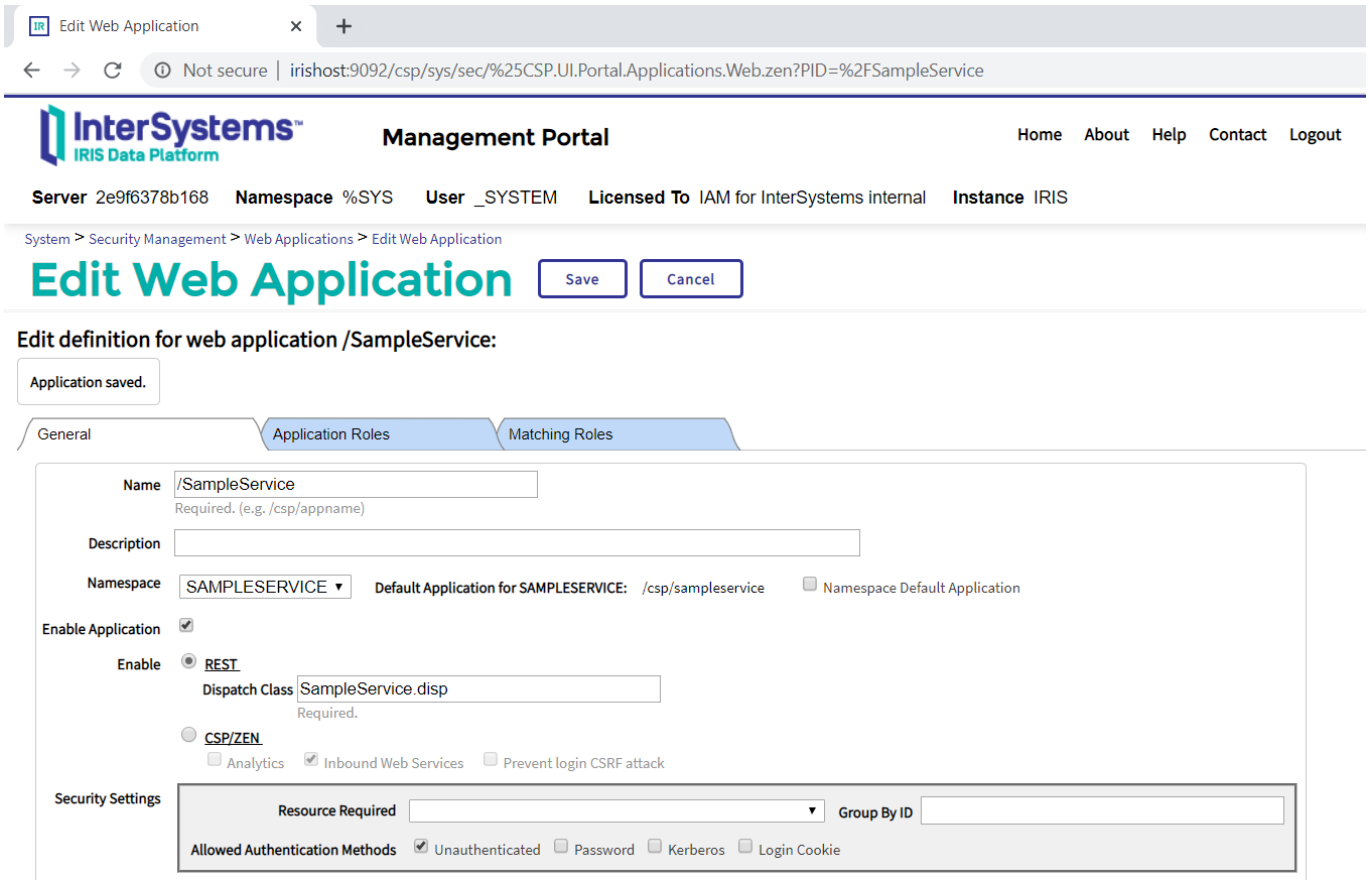
1. O usuário insere as credenciais (por exemplo, nome de usuário e senha) na aplicação cliente
2. A aplicação cliente envia as credenciais do usuário junto com sua própria identificação (id do cliente e segredo do cliente, por exemplo) para o servidor de autorização. O servidor de autorização valida as credenciais do usuário e a identificação do cliente e retorna um token de acesso
3. O cliente usa o token para acessar recursos no servidor de recursos
4. O servidor de recursos valida o token de acesso recebido antes de retornar qualquer informação ao cliente

Com isso em mente, existem dois cenários em que você pode usar o IAM para lidar com o OAuth 2.0:

1. O IAM atuando como um validador, verificando o token de acesso fornecido pela aplicação cliente, encaminhando a solicitação para o servidor de recursos apenas se o token de acesso for válido. Neste caso, o token de acesso seria gerado por um servidor de autorização de terceiros
2. O IAM atua como um servidor de autorização, fornecendo token de acesso ao cliente, e como um validador de token de acesso, verificando o token de acesso antes de redirecionar a solicitação ao servidor de recursos.

Definições de IRIS e IAM

Nesta postagem, será utilizado uma aplicação IRIS Web denominada " /SampleService " . Como você pode ver na captura de tela abaixo, este é um serviço REST não autenticado implantado no IRIS:



Adicionalmente, é configurado um serviço chamado " SampleIRISService " no lado do IAM contendo uma rota, como você pode ver na imagem abaixo:

Além disso, é configurado um consumidor chamado " ClientApp " no IAM, inicialmente sem nenhuma credencial, para identificar quem está chamando a API no IAM:

Com as configurações acima, o IAM faz proxy para o IRIS a cada solicitação GET enviada para a seguinte URL:

http://iamhost:8000/event

Neste ponto, nenhuma autenticação é usada ainda. Portanto, se enviarmos uma solicitação GET simples, sem autenticação, para a URL

http://iamhost:8000/event/1

obtemos a resposta desejada.

Neste artigo, vamos usar uma aplicação chamada " PostMan " para enviar solicitações e verificar as respostas. Na captura de tela abaixo do PostMan, você pode ver a solicitação GET simples junto com sua resposta.

Continue lendo até a segunda parte desta série para entender como configurar o IAM para validar os tokens de acesso presentes nas solicitações de entrada.

#API #OAuth2 #REST API #Segurança #InterSystems IRIS

URL de origem:https://pt.community.intersystems.com/post/protegendo-suas-apis-com-oauth-20-no-intersystems-api-management-parte-1