SERVIDOR DE INTEGRAÇÕES SEPARADO · GUIA PRÁTICO

Do zero ao primeiro método

Como ligar um servidor de integrações à plataforma, ligar uma conta e chamar um método — explicado de princípio, sem pressupor nada.

Passo 0 · o mapa: quem é quem

Passo 1 · gerar as chaves no servidor de integrações

Passo 2 · colar no produto (URL + código) e emparelhar

Passo 3 · ligar a primeira conta (exemplo: Gmail)

Passo 4 · dar acesso a um agente

Passo 5 · chamar um método (o pedido assinado)

Passo 6 · o que fica registado, e como se desliga

Resumo · tudo numa página + glossário



O QUE VAIS PRECISAR

Cinco minutos, uma conta de email para o exemplo, e acesso ao servidor onde o serviço de integrações vai correr. Nada de programação: os únicos campos que se preenchem são um URL e um código de oito caracteres.

Versão 0.1 · 25 de novembro de 2026

MICP · Managed Integration & Credential Protocol

Este documento descreve o que já existe e o que está a ser construído, sem prometer o que não existe.

PASSO 0

O mapa: quem é quem

Há dois servidores. Um sabe **pedir**. O outro sabe **executar**. Podem estar na mesma máquina ou em máquinas diferentes — e é indiferente.

A plataforma

Onde estão os agentes, as pipelines, a interface e os utilizadores.

Sabe pedir: “envia este email”.

⇄

pedidos
respostas

O servidor de integrações

Onde estão o cofre, as contas e os métodos.

Sabe executar: abre o cofre, pega a chave do Gmail e envia.

Fig. 1 — dois papéis distintos. A plataforma nunca vê uma chave de fornecedor.

Porque é que separar ajuda

SE ESTIVEREM JUNTOS	SE ESTIVEREM SEPARADOS
Uma falha na plataforma é uma falha nas credenciais	O cofre vive noutra máquina, com outra política de acesso e outro backup
Só podes ter um	Um cliente pode ligar o servidor dele ao teu produto
Rodar a chave do cofre mexe no produto	Roda-se no servidor, sem tocar na plataforma

A REGRA QUE EVITA O PROBLEMA QUE TEMIAS

O emparelhamento é **manual e sem dependências**: copia-se um URL e um código. Não se cria uma integração para ligar o servidor de integrações — senão uma dependia da outra, e não haveria primeiro passo possível.

> Dois servidores, dois papéis: um pede, o outro executa. Nenhum guarda as chaves do outro.

Passo 1 · gerar as chaves →

PASSO 1

Gerar as chaves no servidor de integrações

Acontece uma vez, na máquina do servidor de integrações, feito por quem o administra. Não é a plataforma que gera nada.

```
$ micp-server init --nome "integrações da loja"

  ✓ chave privada        /etc/micp/server.key      (0400, só root)
  ✓ chave pública        /etc/micp/server.pub      (pode ser partilhada)
  ✓ identificador        srv_7c1a9f2e
  ✓ fingerprint          9f3c-1a2b-7e44-0d18
  ✓ código de emparelhamento   K7QP-3M2X      (válido 10 minutos)
```

Chave privada
Assina o que o servidor responde e decifra o cofre.
nunca sai da máquina

Chave pública
A plataforma usa-a para verificar as respostas do servidor.
feita para ser partilhada

Fingerprint
Para confirmar em voz alta "é mesmo este servidor", sem revelar a chave.
9f3c-1a2b…

Fig. 2 — o par de chaves. A privada fica; a pública viaja.

Porque é um par de chaves e não uma password copiada

Com um par de chaves, cada lado prova a sua identidade sem enviar o segredo: o servidor assina, a plataforma verifica. Uma password partilhada serviria para os dois lados — quem a visse falava pelos dois.

O QUE A PESSOA LEVA PARA O PRODUTO

Duas coisas: o **URL** do servidor e o **código de emparelhamento**. O código vale 10 minutos e serve uma vez. As chaves trocam-se automaticamente no primeiro contacto — não se copiam à mão.

> O servidor cria as chaves e fica com a privada. Daqui saem apenas um URL e um código de oito caracteres.

PASSO 2

Colar no produto e emparelhar

Na plataforma existe um ecrã com dois campos. É o único sítio onde há configuração manual em todo o processo.



Fig. 3 — dois campos. Nada de ficheiros, nada de chaves coladas à mão.

O que acontece quando se carrega em “Ligar”

1. A plataforma chama `GET /micp/manifesto` — “quem és tu?”
2. O servidor responde com o identificador, a versão e a **chave pública** dele.
3. A plataforma envia `POST /micp/parear` com o código e a **chave pública dela**.
4. O servidor confere o código, guarda a chave da plataforma e responde assinado.
5. Cada lado tem agora a chave pública do outro. O código é destruído.

Fig. 4 — emparelhamento em cinco passos. Depois disto, os dois lados reconhecem-se para sempre.

PORQUE NÃO SE COLA A CHAVE PÚBLICA À MÃO

Poderia — mas uma chave de 60 caracteres num campo de texto é uma gralha à espera de acontecer, e uma gralha aqui dá “servidor não reconhecido” sem explicação. O código de 8 caracteres faz o mesmo trabalho e expira.

> Emparelhar é trocar chaves públicas uma vez. Depois disso, os dois lados reconhecem-se sem intervenção.

PASSO 3

Ligar a primeira conta (exemplo: Gmail)

Agora que os servidores se conhecem, liga-se uma conta. Esta parte acontece **no servidor de integrações**, não na plataforma.

1. **Escolher a integração:** `gmail` — já vem com os métodos (`send` , `read`).
2. **Dar um nome à conta:** `elizeu` . É o que se usa na chamada.
3. **Pedir os valores:** o email e a palavra-passe de aplicação, com o link exacto onde a obter.
4. **Guardar no cofre:** cifrado com a chave do cofre. Fica só o ciphertext.
5. **Testar a ligação:** uma chamada real. Só fica ligada se responder.

Fig. 5 — ligar uma conta. O passo 5 é o que separa “configurado” de “a funcionar”.

O que fica guardado depois deste passo

ONDE	O QUE
Servidor de integrações	a conta `gmail · elizeu` e a credencial cifrada no cofre
Plataforma	nada — ainda não sabe que esta conta existe

A plataforma só descobre as contas quando perguntar — e só vê as que lhe forem autorizadas. É o passo seguinte.

> Ligar a conta é pedir, cifrar e testar. Sem teste verde, a conta não serve para nada.

PASSO 4

Dar acesso a um agente

Ter a conta ligada não chega: é preciso dizer **quem** a pode usar e **para quê**.

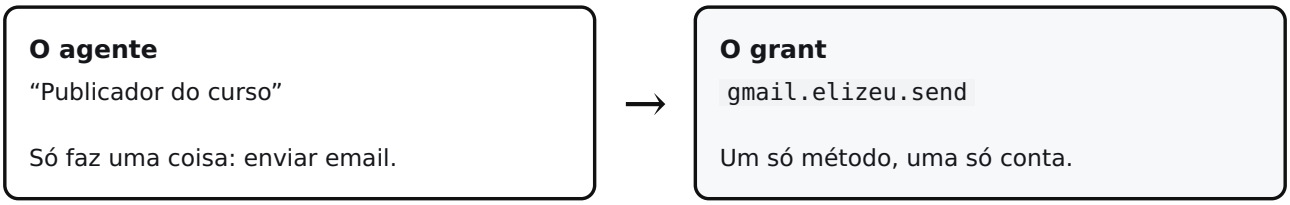

Fig. 6 — o grant liga um agente a um método concreto de uma conta concreta.

As três precisões possíveis

ESCREVE-SE	DÁ ACESSO A
`gmail.elizeu.send`	só esse método, só essa conta
`gmail.elizeu.*`	todos os métodos dessa conta
`gmail.*`	todos os métodos de todas as contas Gmail

O QUE O AGENTE PASSA A VER

Apenas o que lhe foi concedido. A lista que ele consulta é **filtrada** — o que não pode usar não aparece. Não é uma recusa em execução: é invisibilidade. Um agente não tenta o que não vê.

> Um grant é uma frase: este agente, este método, esta conta.

PASSO 5

Chamar um método (o pedido assinado)

Agora o agente pede. O pedido vai **assinado**: a assinatura prova que aqueles bytes são mesmo dele e não foram alterados.

```
POST /methods/gmail.send
Authorization: Bearer ***
X-Micp-Run:  r_8f21
X-Micp-Ts:   1795649100
X-Micp-Nonce: 7c1a9f

{"account":"elizeu","to":"joao@exemplo.pt","subject":"Relatório","body":"Segue em anexo."}
```

1. **Verifica a assinatura** — com o segredo daquela execução. Se não bater, recusa.
2. **Verifica o tempo e o nonce** — pedido com mais de 5 minutos, ou repetido, é recusado.
3. **Confere o escopo** — este chamador pode `gmail.elizeu.send`? (o grant, não a assinatura)
4. **Abre o cofre** — decifra a credencial do Gmail, só durante o pedido.
5. **Executa e responde** — `{"sent":true,"id":"…"}`, e registra tudo.

Fig. 7 — cinco verificações antes de tocar no segredo. Nenhuma delas acontece na plataforma.

DUAS PERGUNTAS DIFERENTES, DOIS MECANISMOS

A **assinatura** responde “é mesmo ele e não foi alterado?”. O **escopo** responde “pode fazer isto?”. Confundir as duas é o erro clássico: fica-se com autenticação sem autorização.

> A assinatura diz quem é; o escopo diz se pode. As duas são verificadas antes de tocar no segredo.

PASSO 6

O que fica registado e como se desliga

Registo de cada chamada

conta · método · agente · execução · resultado · duração · hora

Nunca: o segredo, nem o corpo que o possa conter.

Desligar

Revogar um grant → o agente deixa de ver o método, de imediato.
Apagar a conta → todas as chamadas dessa conta falham.

Fig. 8 — o registo é o que permite responder “o que é que os agentes fizeram com as minhas credenciais?”.

Rodar a chave do cofre (sem parar nada)

1
Gera-se a chave v2. As duas coexistem.

→

2
As linhas antigas abrem com v1 e são re-cifradas em fundo.

→

3
Quando não sobra nenhuma v1, a chave v1 é destruída.

Fig. 9 — rotação. Ninguém volta a colar chaves; nada fica indisponível a meio.

O TESTE QUE NINGUÉM FAZ

Restaurar a base de dados **e** a chave do cofre (de sítios diferentes) e provar que uma entrada abre. Sem esse ensaio, um backup é uma esperança.

> Registar tudo é o que permite responder a um cliente em cinco minutos, com factos e não com memória.

RESUMO

Tudo numa página

PASSO	O QUE ACONTECE	ONDE
0	Dois papéis: um pede, o outro executa	—
1	Gerar chaves privada/pública + código de emparelhamento	servidor de integrações
2	Colar URL + código; os dois lados trocam chaves públicas	plataforma (2 campos)
3	Ligar a conta, guardar no cofre, testar	servidor de integrações
4	Conceder `gmail.elizeu.send` ao agente	plataforma
5	Pedido assinado → 5 verificações → executa	entre os dois
6	Registo, revogação, rotação, teste de restauro	servidor de integrações

Glossário

chave do cofre	a única chave nossa. Cifra os segredos. Não decide permissões.
cofre	onde ficam os segredos, cifrados, dentro da base de dados.
conta	uma credencial ligada: `gmail · elizeu`.
método	uma operação: `send`, `read`. Definida uma vez, no tipo.
grant	“o agente X pode usar o método Y da conta Z”.
passe de execução	autorização temporária (15 min) de uma execução concreta.
assinatura	prova de que os bytes são de quem diz e não foram alterados.
emparelhar	trocar chaves públicas uma vez, com um código que expira.

A frase que resume os sete passos: **o servidor de integrações tem as chaves; a plataforma tem os agentes; o agente pede, assinado; o servidor verifica, decifra, executa e registra.**