



トレース API

スクリプトのトレーシングにより、スクリプト作成者がスクリプト内からトレースを生成することが可能になります。これは、スクリプトのデバッグだけに使用してください。スクリプトを稼働させ始めるときには、無効にする必要があります。スクリプトのトレーシングを有効にする方法は2通りあります。

1. 設定を使用する：SIP トランクに関連した [Script Trace] チェックボックスを参照してください。
2. スクリプトを使用する：`trace.enable()` API を参照してください。

次の表に、上述の2つの設定のすべての組み合わせにおいて、トレースがスクリプトによって生成されるかどうかを示します。

設定	スクリプト	トレースの生成
有効	有効	あり
有効	無効	なし
無効	有効	なし
無効	無効	なし

トレース ライブラリには、次の API があります。

- [「trace.format」](#)
- [「trace.enable」](#)
- [「trace.disable」](#)
- [「trace.enabled」](#)

trace.format

```
trace.format(format-string, p1, p2, ...)
```

この関数は、フォーマット文字列と一連のパラメータを受け取って、スクリプトに関連付けられているトランクのトレーシングが有効になっており、スクリプトが `trace.enable()` によってトレーシングを有効にしていた場合に、パラメータをフォーマット文字列に挿入し、その出力を SDI トレース ファイルに書き出します。

例

```
M = {}
trace.enable()
function M.inbound_INVITE(msg)
    local callid = msg.getHeader("Call-Id")
    trace.format("Call-Id header is %s", callid)
end
return M
```

trace.enable

```
trace.enable()
```

この関数は、スクリプトがトレーシングをローカルに有効にすることを可能にします。



(注) トレーシングは、そのスクリプトを使用する SIP トランクの設定でも有効になっている必要があります。

例

前述の例、「[trace.format](#)」を参照してください。

trace.disable

```
trace.disable()
```

この関数は、スクリプトがトレーシングをローカルに無効にすることを可能にします。



(注) 関連付けられているトランクのスクリプト トレース設定フラグがオンになっている場合は、たとえここで無効にしても、`Trace.output` によってスクリプト トレースが生成されます。

例 :

```
M = {}
trace.disable()
function M.inbound_INVITE(msg)
    local callid = msg.getHeader("Call-Id")
    trace.format("Call-Id header is %s", callid)
end
return M
```

trace.enabled

```
trace.enabled() turns a boolean
```

スクリプトのこのインスタンスに関連付けられているデバイス上でスクリプト トレーシングが有効になっているか、またはスクリプト自身でトレーシングを有効にしていた場合に、`true` が返されます。トレーシングが有効になっていない場合は、`false` が返されます。

例

```
M = {}
trace.enable()
function M.inbound_INVITE(msg)
    if trace.enabled()
    then
        local callid = msg.getHeader("Call-Id")
        trace.format("Call-Id header is %s", callid)
    end
end
return M
```