Direction under regulation 2(2) of the Delivery of Tax Information through Software (Ancillary Metadata) Regulations 2019 (S.I. 2019/360)

Introduction

HMRC has legal duties to keep taxpayers’ information confidential and to administer and collect the taxes and duties due to the Exchequer.

A key security approach HMRC takes to protect customer data is transaction monitoring. Transaction monitoring refers to data collected through electronic communications with HMRC which we use to protect customer data and guard against fraud. It is an approach adopted globally by governments and non-governmental organisations. HMRC’s approach is in line with National Cyber Security Centre (NCSC) and Cabinet Office recommended guidance and industry good practice.

Making Tax Digital for Value Added Tax (VAT) begins on 1 April 2019. This affects VAT registered taxpayers with a taxable turnover at or above the VAT registration threshold. The law will require these businesses to use functional compatible software to keep certain records in electronic form and submit their VAT returns digitally. These taxpayers will use commercial software to meet some or all of these requirements. Information will be transmitted to and from HMRC using an Application Programming Interface (API). Taxpayers may also choose to use API-enabled software for the purposes of meeting VAT or other tax obligations even if not required to do so by law.

Without the protection offered by transaction monitoring, this data could be compromised, leading to fraud against taxpayers or the UK Exchequer. HMRC needs to consistently and reliably receive fraud header information to ensure this protection takes place efficiently and effectively. HMRC stores this information securely.

HMRC has made regulations under section 135 Finance Act 2002. These regulations place a legal obligation on software suppliers. This will ensure their API-enabled products capture and transmit all available fraud header information along with taxation information that it relates to. This information must be sent to HMRC systems electronically from the products that taxpayers use. Not carrying out this legal obligation risks a penalty and also having access to HMRC systems removed. HMRC will work with a software developer to rectify any non-compliance before seeking to impose a penalty.
The penalty would not exceed £3,000 and be limited to one penalty per year in relation to any one product of a software developer. However HMRC will have discretion over when to assess the penalty. Software developers will have the right to appeal against a penalty if they believe it has been imposed incorrectly.

**Header Information**

“Header information” is a list of metadata requirements which developers need to build in to their software to capture fraud prevention data. This information is captured when electronic systems exchange data, for example between software and HMRC. This data is collected by HMRC – this process is known as transaction monitoring.

The Commissioners for Her Majesty’s Revenue and Customs give the following direction under regulation 2(2) of the Delivery of Tax Information through Software (Ancillary Metadata) Regulations 2019 (S.I. 360/2019) (“the regulations”).

The direction defines a set of “relevant ancillary metadata” for the purposes of the regulations.

The direction comes into force on 1st April 2019.

---

**The required header information**

The obligation below to provide header information applies to relevant software programs using the following API: VAT (MTD) (see [https://developer.service.hmrc.gov.uk/api-documentation/docs/api/service/vat-api/1.0](https://developer.service.hmrc.gov.uk/api-documentation/docs/api/service/vat-api/1.0)) from 1st April 2019.

The header information necessary for the purpose of ensuring the authenticity and security of a delivery of tax information (as defined in Reg 1(2) of the regulations) to the Commissioners through relevant software (as defined in Reg 1(2) of the regulations) is as follows:

**Header contents**

Header contents must use the character set US-ASCII, with other characters percent encoded as in [RFC3986](https://www.rfc-editor.org/rfc/rfc3986) (opens in a new tab). Other restrictions on header values are shown under the individual header definitions.

The application connection method determines which headers we require:

- **MOBILE_APP_DIRECT**
  - Installed mobile application connecting directly to HMRC
**Header guidance**

**Originating device**

*Originating device* means the device that initiates an action.

In most connection methods, it’s most likely the device that is physically in the possession of the person initiating the requests.

In **BATCH_PROCESS_DIRECT**, it might be a vendor’s server.

In **DESKTOP_APP_DIRECT** or **DESKTOP_APP_VIA_SERVER**, it might be a hosted desktop environment accessed remotely by the user.

Where multiple vendors are involved in managing the intermediary servers between a client and HMRC, the header contents must reflect information from all vendors.
Headers not required

If a header is not required for your connection method, then you can leave it out entirely when making your submission, or submit it with an empty header value. You must not supply null values, for example, the following are not valid:

- Gov-Client-User-IDs: null
- Gov-Client-User-IDs: nil
- Gov-Client-User-IDs: undefined

Header values cannot be collected

In some cases, while the header itself is required for your connection method, we anticipate you might not be able to collect a value because:

- of operating system or platform restrictions beyond your reasonable control
- of security measures beyond your reasonable control
- the header is not applicable, as called out explicitly by the header documentation

Only in these cases, you must submit the header with an empty value or omit it entirely. You must not supply null values.

In all other cases, you must submit a valid value and not leave the header empty.

For example, an application connecting directly to HMRC might not be able to supply a username for Gov-Client-User-IDs if the operating system of the originating device has no concept of user accounts, so the header can be omitted entirely or left empty as follows:

- Gov-Client-User-IDs:

Key-value encoding

Whenever a header contains a key-value data structure, it must be in the form:

- `<key-1>=<value-1>&<key-2>=<value-2>&...`
Whenever a key is applicable but has no applicable value, the key-value pair can be omitted, or the key can be included with an empty value.

Keys and values must be percent encoded as in RFC3986 (https://tools.ietf.org/html/rfc3986#section-2.1).

The key-value pairs can be given in any order.

**List encoding**

Whenever a header contains a list, it must be of the form:

```
<value-1>,<value-2>,...
```

Values must be percent encoded as in RFC3986 (https://tools.ietf.org/html/rfc3986#section-2.1).

Values must not be empty.

**Header definitions**

**Gov-Client-Connection-Method**

A string representing the connection method used for the request.

One of:

- MOBILE_APP_DIRECT
- DESKTOP_APP_DIRECT
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- WEB_APP_VIA_SERVER
- BATCH_PROCESS_DIRECT
- OTHER_DIRECT
- OTHER_VIA_SERVER

For example
**Gov-Client-Connection-Method**: DESKTOP_APP_VIA_SERVER

---

**Gov-Client-Public-IP**

The public IP address (IPv4 or IPv6) from which the originating device makes the request.

**Required for connection methods:**
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- WEB_APP_VIA_SERVER
- OTHER_VIA_SERVER

**For example:**
- Gov-Client-Public-IP: 198.51.100.0

---

**Gov-Client-Public-Port**

The public TCP port that the originating device uses when initiating the request.

**Required for connection methods:**
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- WEB_APP_VIA_SERVER
- OTHER_VIA_SERVER

**For example:**
- Gov-Client-Public-Port: 12345

---

**Gov-Client-Device-ID**

An identifier unique to an originating device. This should be generated by an application and persistently stored on the device. The identifier should not expire.
If an identifier cannot be persistently stored due to security restrictions, submit the header with an empty value or omit it entirely.

**Required for connection methods:**

MOBILE_APP_DIRECT
DESKTOP_APP_DIRECT
MOBILE_APP_VIA_SERVER
DESKTOP_APP_VIA_SERVER
WEB_APP_VIA_SERVER
BATCH_PROCESS_DIRECT
OTHER_DIRECT
OTHER_VIA_SERVER

**For example:**

Gov-Client-Device-ID: beec798b-b366-47fa-b1f8-92cede14a1ce

**Gov-Client-User-IDs**

A key-value data structure containing the user identifiers. The keys should indicate accounts the user holds.

The `os` field should contain the identifier of the user signed in on the originating device. Additional fields should contain the user’s identifiers with the vendor services involved in the request.

If a user identifier cannot be accessed due to operating system permissions or security controls, then it can be left blank or omitted.

**Required for connection methods:**

MOBILE_APP_DIRECT
DESKTOP_APP_DIRECT
MOBILE_APP_VIA_SERVER
DESKTOP_APP_VIA_SERVER
WEB_APP_VIA_SERVER
BATCH_PROCESS_DIRECT
OTHER_DIRECT
OTHER_VIA_SERVER
**For example:**

In **DESKTOP_APP_VIA_SERVER** connection method, when there is a single vendor service involved in handling the request:

Gov-Client-User-IDs: os=alice_desktop&my-vendor-online-account=alice_online_account_user_id_with_vendor

For example, in **DESKTOP_APP_VIA_SERVER** connection method, when the request passes through two vendor services:

Gov-Client-User-IDs: os=alice_desktop&my-vendor=alice_online_account_user_id_with_vendor&my-secondary-vendor=alice_online_account_user_id_with_secondary_vendor

For example, in **DESKTOP_APP_DIRECT**, where there are no vendor accounts involved, or **BATCH_PROCESS_DIRECT** where there are no user devices involved and the originating device is the batch server:

Gov-Client-User-IDs: os=user123

**Gov-Client-Timezone**

The local timezone of the originating device, expressed as UTC±<hh>:<mm>.

**Required for connection methods:**

- **MOBILE_APP_DIRECT**
- **DESKTOP_APP_DIRECT**
- **MOBILE_APP_VIA_SERVER**
- **DESKTOP_APP_VIA_SERVER**
- **WEB_APP_VIA_SERVER**
- **BATCH_PROCESS_DIRECT**
- **OTHER_DIRECT**
- **OTHER_VIA_SERVER**

**For example:**

Gov-Client-Timezone: UTC+01:00
Gov-Client-Timezone: UTC-01:15
Gov-Client-Local-IPs

A list of all local IP addresses (IPv4 and IPv6) available to the originating device. If local device IP addresses cannot be discovered – for example, through a web browser due to permissions or lack of WebRTC – then submit the header with an empty value or omit it entirely.

Required for connection methods:

- MOBILE_APP_DIRECT
- DESKTOP_APP_DIRECT
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- WEB_APP_VIA_SERVER
- BATCH_PROCESS_DIRECT
- OTHER_DIRECT
- OTHER_VIA_SERVER

For example:

Gov-Client-Local-IPs: 10.1.2.3,10.3.4.2

Gov-Client-Screens

Information related to the originating device’s screens. The fields include:

- width: is the reported width of the screen, in pixels
- height: is the reported height of the screen, in pixels
- scaling-factor: is the reported scaling factor of the screen. For instance, high pixel density screens might have a scaling factor of 2, whereas standard definition screens might report a scaling factor of 1. If the scaling factor cannot be retrieved, leave this empty or omit it entirely.
- colour-depth: is the colour depth of the screen, in bits. If the colour depth cannot be retrieved, leave this empty or omit it entirely.
Required for connection methods:

MOBILE_APP_DIRECT
DESKTOP_APP_DIRECT
MOBILE_APP_VIA_SERVER
DESKTOP_APP_VIA_SERVER
WEB_APP_VIA_SERVER

For example:
Gov-Client-Screens: width=1920&height=1080&scaling-factor=1&colour-depth=16
width=3000&height=2000&scaling-factor=1.25&colour-depth=

Gov-Client-Window-Size

The number of pixels of the window on the originating device in which the user initiated (directly or indirectly) the API call to HMRC.

If the call was initiated by a background process, submit the header with an empty value or omit it entirely.

Should be displayed as a key-value data structure with width and height fields.

Required for connection methods:

MOBILE_APP_DIRECT
DESKTOP_APP_DIRECT
MOBILE_APP_VIA_SERVER
DESKTOP_APP_VIA_SERVER
WEB_APP_VIA_SERVER

For example:
Gov-Client-Window-Size: width=1256&height=803

Gov-Client-User-Agent

An attempt to identify the operating system family, version, device manufacturer and model of the originating device.

Reported in the format:
<table>
<thead>
<tr>
<th>OS Family/OS Version+ (Device Manufacturer/Device Model+)</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you cannot detect the exact version due to operating system constraints, but can detect several options, list all options. Each value (OS Family, OS Version, Device Manufacturer and Device Model) must be percent encoded but not the separators themselves. The separators are / ( ).</td>
</tr>
</tbody>
</table>

**Required for connection methods:**

- MOBILE_APP_DIRECT
- DESKTOP_APP_DIRECT
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- BATCH_PROCESS_DIRECT
- OTHER_DIRECT
- OTHER_VIA_SERVER

**For example:**

Gov-Client-User-Agent: Windows/XP Windows/NT (Dell/XPS15 Dell/XPS13)

If you cannot discover any information due to operating system constraints, leave the appropriate field blank as shown in the following examples:

Gov-Client-User-Agent: Windows/ (Dell/XPS15)
Gov-Client-User-Agent: / (Dell/XPS15)
Gov-Client-User-Agent: Windows/XP (/)
Gov-Client-User-Agent: Windows/XP (/XPS15)
Gov-Client-User-Agent: Windows/XP Windows/98 (/)

**Gov-Client-Browser-Plugins**

A list of browser plugins on the originating device. If none can be discovered, due to browser restrictions or the lack of installed plugins, submit the header with an empty value or omit it entirely.

**Required for connection methods:**

- WEB_APP_VIA_SERVER
For example:
Gov-Client-Browser-Plugins: Shockwave%20Flash,Chromium%20PDF%20Viewer

Gov-Client-Browser-JS-User-Agent

JavaScript-reported user agent string from the originating device.

Required for connection methods:
WEB_APP_VIA_SERVER

For example:
Gov-Client-Browser-JS-User-Agent: Mozilla/5.0 (iPad; U; CPU OS 3_2_1 like Mac OS X; en-us) AppleWebKit/531.21.10 (KHTML, like Gecko) Mobile/7B405

Gov-Client-Browser-Do-Not-Track

Whether the Do Not Track option is enabled on the browser.

It should be true if this option is enabled, false if it’s not. If it cannot be collected because the browser does not have or expose this feature, submit the header with an empty value or omit it entirely.

Required for connection methods:
WEB_APP_VIA_SERVER

For example:
Gov-Client-Browser-Do-Not-Track: false

Gov-Client-Multi-Factor

A list of key-value data structures containing details of the multi-factor authentication (MFA) statuses related to the API call. For each key-value data structure, the applicable fields are:

type is a tag describing the MFA method being used. The possible values are:
• **TOTP** - The MFA was performed by accepting a time-based one-time password code

• **AUTH_CODE** - The MFA was performed by sending an authorisation code to the user in some out-of-band channel, for example, by email or by SMS

• **OTHER** - A different MFA method was used than the available values

**timestamp** is a UTC timestamp in the format yyyy-MM-ddThh:mmZ recording the time of the last successful prompt for this factor.

**unique-reference** identifies a single factor. For example, a salted-and-hashed phone number used for SMS or an identifier linked to a TOTP secret – but not the secret itself. The intention is to recognise the same factor being used across API calls.

If only a single factor (for example, username and password) is being used, submit the header with an empty value or omit it entirely.

**Required for connection methods:**

- MOBILE_APP_DIRECT
- DESKTOP_APP_DIRECT
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- WEB_APP_VIA_SERVER
- OTHER_DIRECT
- OTHER_VIA_SERVER

**For example:**


**Gov-Vendor-Version**

A key-value data structure of software versions involved in handling a request:

```
<software-name>=<version-number>&<software-name-2>=<version-number-2>& ...
```
**Required for connection methods:**

- MOBILE_APP_DIRECT
- DESKTOP_APP_DIRECT
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- WEB_APP_VIA_SERVER
- BATCH_PROCESS_DIRECT
- OTHER_DIRECT
- OTHER_VIA_SERVER

**For example:**

For **DESKTOP_APP_DIRECT**, the header may be:

```
Gov-Vendor-Version: my-desktop-software=1.2.3.build4286
```

For **MOBILE_APP_VIA_SERVER**, the header may be:

```
Gov-Vendor-Version: my-frontend-app=2.2.2&my-serverside-code=v3.8
```

**Gov-Vendor-License-IDs**

A key-value data structure of hashed license keys relating to the vendor software initiating the API request on the originating device.

```
<software-name>=<hashed-license-value>&<software-name-2>=<hashed-license-value-2>& ...
```

If there are no such licenses on the originating device, then submit the header with an empty value or omit it entirely.

**Required for connection methods:**

- MOBILE_APP_DIRECT
- DESKTOP_APP_DIRECT
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- WEB_APP_VIA_SERVER
- BATCH_PROCESS_DIRECT
For example:
Gov-Vendor-License-IDs: my-licensed-software=8D7963490527D33716835EE7C195516D5E562E03B224E9B359836466EE40CDE1

Gov-Vendor-Public-IP

The public IP address of the servers to which the originating device sent their requests. This might be an IP address of a Web Application Firewall, a DDoS Protection Service, or a load balancer that the vendor's DNS record resolves to.

Required for connection methods:
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- WEB_APP_VIA_SERVER
- OTHER_VIA_SERVER

For example:
Gov-Vendor-Public-IP: 203.0.113.6

Gov-Client-MAC-Addresses

The list of MAC addresses available on the originating device. If you cannot retrieve them due to operating system restrictions, then submit the header with an empty value or omit it entirely.

Required for connection methods:
- MOBILE_APP_DIRECT
- DESKTOP_APP_DIRECT
- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- BATCH_PROCESS_DIRECT
- OTHER_DIRECT
**OTHER_VIA_SERVER**

**For example:**

Gov-Client-MAC-Addresses:

\[\text{ea%3A43%3A1a%3A5d%3A21%3A45,10%3A12%3Acc%3Afa%3Aa%3A32}\]

**Gov-Vendor-Forwarded**

A list that details hops over the internet between services that terminate TLS.

For each hop over the internet, a key-value data structure with a *by* and *for* field must be appended to the list.

The *by* field must be the server’s public IP address where it received the request.

The *for* field must be the requestor’s public IP address from which the vendor received the request.

**Required for connection methods:**

- MOBILE_APP_VIA_SERVER
- DESKTOP_APP_VIA_SERVER
- WEB_APP_VIA_SERVER
- OTHER_VIA_SERVER

**For example, if:**

1. a request is first received by a web application firewall (WAF) service on 33.252.57.234. The WAF sees the request coming from 57.4.28.41. Subsequently, the WAF passes the request on to the vendor servers via HTTPS over the internet.
2. the request is received by the server of vendor A on 188.87.76.95, from the WAF-owned IP address 209.210.136.84
3. The services of vendor A issue a HTTP request as a result of the previous step to the services of vendor B on 176.30.57.118, from vendor A owned IP 150.94.192.63

then you should generate the following header:
Angela MacDonald

Melissa Tatton

Two of the Commissioners for Her Majesty’s Revenue and Customs

Date: 27th March 2019

---
