

User-Agent header - HTTP | MDN

Published: 2025-10-28 · Archived: 2026-04-05 17:50:47 UTC

Syntax

Common format for web browsers:

Directives

`<product>`

A product identifier — its name or development codename.

`<product-version>`

Version number of the product.

`<comment>`

Zero or more comments containing more details. For example, sub-product information.

User-Agent reduction

The information exposed in the `User-Agent` header has historically raised [privacy](#) concerns — it can be used to identify a particular user agent, and can therefore be used for [fingerprinting](#). To mitigate such concerns, [supporting browsers](#) provide a reduced set of information in their `User-Agent` header, and in related API features such as [Navigator.userAgent](#), [Navigator.appVersion](#), and [Navigator.platform](#).

For example, whereas previously the `User-Agent` string for Chrome running on Android might have looked like this:

```
Mozilla/5.0 (Linux; Android 16; Pixel 9) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.12.45 Mobile Safari/537.36
```

After the User-Agent reduction update, it now looks like this:

```
Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Mobile Safari/537.36
```

- The platform version is always a fixed value, in this case, `Android 10`.
- The device model is always a fixed value, in this case, `K`.
- The Chrome major version number shows correctly, but the minor version numbers are always shown as zeros — `0.0.0`.

Servers that need more information can request it via [User-Agent client hints](#). After the initial connection, the server can send an `Accept-CH` response header detailing the data items they want, and the client can then send the data back via `Sec-CH-UA-*` headers. This information can also be accessed via the [User-Agent Client Hints API](#).

For more detailed information, including a guide to retrieving more information as required, see [User-Agent reduction](#). You can also find examples of reduced `User-Agent` strings in the following sections.

Firefox UA string

For more on Firefox- and Gecko-based user agent strings, see the [Firefox user agent string reference](#). The UA string of Firefox is broken down into 4 components:

```
Mozilla/5.0 (platform; rv:gecko-version) Gecko/gecko-trail Firefox/firefox-version
```

1. `Mozilla/5.0` is the general token that says that the browser is Mozilla-compatible. For historical reasons, almost every browser today sends it.
2. **platform** describes the native platform that the browser is running on (Windows, Mac, Linux, Android, etc.) and if it is a mobile phone. Note that **platform** can consist of multiple `;`-separated tokens. See below for further details and examples.
3. **rv:gecko-version** indicates the release version of Gecko (such as "17.0"). In recent browsers, **gecko-version** is the same as **firefox-version**.
4. **Gecko/gecko-trail** indicates that the browser is based on Gecko. (On the desktop, **gecko-trail** is always the fixed string `20100101`.)
5. **Firefox/firefox-version** indicates that the browser is Firefox and provides the version (such as "17.0").

Desktop examples:

```
Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:47.0) Gecko/20100101 Firefox/47.0
```

```
Mozilla/5.0 (Macintosh; Intel Mac OS X x.y; rv:42.0) Gecko/20100101 Firefox/42.0
```

Chrome UA string

The Chrome (or Chromium/Blink-based engines) user agent string is similar to Firefox's. For compatibility, it adds strings like `KHTML`, like Gecko and `Safari`. It adds `"CriOS/<version>"` on iPhone.

Desktop examples:

```
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0
```

```
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Sa
```

Android phone example:

```
Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Mobile Sa
```

[Opera UA string](#)

The Opera browser is also based on the Blink engine, which is why it almost looks the same as the Chrome UA string, but adds "OPR/<version>" on desktop and Android, and "OPT/<version>" on iPhone. For preview versions, Opera also includes a description of the particular browser edition in parentheses, for example (Edition developer) .

Desktop examples:

```
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/139.0.0.0
```

```
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/139.0.0.0 Sa
```

Android phone example:

```
Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/140.0.0.0 Mobile Sa
```

[Microsoft Edge UA string](#)

The Edge browser is also based on the Blink engine. It adds "Edg/<version>" on desktop platforms, "EdgA/<version>" on Android, and "EdgiOS/<version>" on iPhone.

Desktop examples:

```
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0
```

```
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Sa
```

Android phone example:

```
Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/141.0.0.0 Mobile Sa
```

[Safari UA string](#)

Safari is based on the WebKit engine, but its UA string is also similar to the Blink-based browsers. It tends to include a Version/xxx string before the actual engine build version to indicate the browser release version, which unlike Blink-based browsers is different. In the case of iPhone (Mobile) Safari, the string also includes Mobile .

Note: At the time of writing, non-Apple iPhone browsers (such as Firefox, Chrome, and Edge) are still based on WebKit, therefore their UA strings are similar to the Safari UA string.

Desktop example:

```
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/26.0
```

iPhone example:

```
Mozilla/5.0 (iPhone; CPU iPhone OS 18_6 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Vers
```

[Pre-user-agent reduction examples](#)

This section provides some examples of UA strings in earlier browser versions, prior to the introduction of user-agent reduction:

Google Chrome:

```
Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.103 Safari/5
```

Microsoft Edge:

```
Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.12
```

Opera:

```
Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.106 Safari/5
```

Older, Presto-based Opera releases used a structure like this:

```
Opera/9.80 (Macintosh; Intel Mac OS X; U; en) Presto/2.2.15 Version/10.00
```

```
Opera/9.60 (Windows NT 6.0; U; en) Presto/2.1.1
```

[Crawler and bot UA strings](#)

[Examples](#)

```
Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
```

```
Mozilla/5.0 (compatible; YandexAccessibilityBot/3.0; +http://yandex.com/bots)
```

Examples

```
curl/7.64.1
```

```
PostmanRuntime/7.26.5
```

Specifications

Specification
HTTP Semantics # field.user-agent

Browser compatibility

See also

- [User-Agent detection, history and checklist](#)
- [Firefox user agent string reference](#)
- [Browser detection using the user agent](#)
- [Client hints](#)

Source: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Reference/Headers/User-Agent>