

**NAME**

xcb\_xf86dri\_auth\_connection -

**SYNOPSIS**

```
#include <xcb/xf86dri.h>
```

**Request function**

```
xcb_xf86dri_auth_connection_cookie_t xcb_xf86dri_auth_connection(xcb_connection_t *conn,
    uint32_t screen, uint32_t magic);
```

**Reply datastructure**

```
typedef struct xcb_xf86dri_auth_connection_reply_t {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint32_t authenticated;
} xcb_xf86dri_auth_connection_reply_t;
```

**Reply function**

```
xcb_xf86dri_auth_connection_reply_t *xcb_xf86dri_auth_connection_reply(xcb_connection_t *conn,
    xcb_xf86dri_auth_connection_cookie_t cookie, xcb_generic_error_t **e);
```

**REQUEST ARGUMENTS**

*conn*           The XCB connection to X11.

*screen*         TODO: NOT YET DOCUMENTED.

*magic*           TODO: NOT YET DOCUMENTED.

**REPLY FIELDS**

*response\_type*   The type of this reply, in this case *XCB\_XF86DRI\_AUTH\_CONNECTION*. This field is also present in the *xcb\_generic\_reply\_t* and can be used to tell replies apart from each other.

*sequence*        The sequence number of the last request processed by the X11 server.

*length*          The length of the reply, in words (a word is 4 bytes).

*authenticated* TODO: NOT YET DOCUMENTED.

**DESCRIPTION****RETURN VALUE**

Returns an *xcb\_xf86dri\_auth\_connection\_cookie\_t*. Errors have to be handled when calling the reply function *xcb\_xf86dri\_auth\_connection\_reply*.

If you want to handle errors in the event loop instead, use *xcb\_xf86dri\_auth\_connection\_unchecked*. See **xcb-requests(3)** for details.

**ERRORS**

This request does never generate any errors.

**SEE ALSO****AUTHOR**

Generated from xf86dri.xml. Contact [xcb@lists.freedesktop.org](mailto:xcb@lists.freedesktop.org) for corrections and improvements.