

**NAME**

xcb\_selinux\_get\_property\_data\_context -

**SYNOPSIS**

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

**Request function**

```
xcb_selinux_get_property_data_context_cookie_t
    xcb_selinux_get_property_data_context(xcb_connection_t *conn, xcb_window_t window,
    xcb_atom_t property);
```

**Reply datastructure**

```
typedef struct xcb_selinux_get_property_data_context_reply_t {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint32_t context_len;
    uint8_t pad1[20];
} xcb_selinux_get_property_data_context_reply_t;
```

**Reply function**

```
xcb_selinux_get_property_data_context_reply_t
    *xcb_selinux_get_property_data_context_reply(xcb_connection_t *conn,
    xcb_selinux_get_property_data_context_cookie_t cookie, xcb_generic_error_t **e);
```

**Reply accessors**

```
char *xcb_selinux_get_property_data_context_context(const
    xcb_selinux_get_property_data_context_request_t *reply);
```

```
int xcb_selinux_get_property_data_context_context_length(const
    xcb_selinux_get_property_data_context_reply_t *reply);
```

```
xcb_generic_iterator_t xcb_selinux_get_property_data_context_context_end(const
    xcb_selinux_get_property_data_context_reply_t *reply);
```

**REQUEST ARGUMENTS**

*conn*            The XCB connection to X11.

`xcb_selinux_get_property_data_context(3)`    XCB Requests    `xcb_selinux_get_property_data_context(3)`

*window*            TODO: NOT YET DOCUMENTED.

*property*          TODO: NOT YET DOCUMENTED.

## REPLY FIELDS

*response\_type*    The type of this reply, in this case `XCB_SELINUX_GET_PROPERTY_DATA_CONTEXT`. 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).

*context\_len*        TODO: NOT YET DOCUMENTED.

## DESCRIPTION

### RETURN VALUE

Returns an `xcb_selinux_get_property_data_context_cookie_t`. Errors have to be handled when calling the reply function `xcb_selinux_get_property_data_context_reply`.

If you want to handle errors in the event loop instead, use `xcb_selinux_get_property_data_context_unchecked`. See **xcb-requests(3)** for details.

## ERRORS

This request does never generate any errors.

## SEE ALSO

### AUTHOR

Generated from `xselinux.xml`. Contact `xcb@lists.freedesktop.org` for corrections and improvements.