TLS "Final" value.

### TLS MAC

The TLS MAC mechanisms are used to generate integrity tags for the TLS "finished" message.

**CKM\_TLS10\_MAC\_SERVER**

**CKM\_TLS10\_MAC\_CLIENT**

**CML\_TLS12\_MAC**

**CKM\_TLS10\_MAC\_SERVER** and **TLS10\_MAC\_CLIENT** take no parameters. They both calculate the verification data as described for TLS1.0 and 1.1 with **CKM\_TLS10\_MAC\_SERVER** using "server finished" as the label, and **CKM\_TLS10\_MAC\_CLIENT** using "client finished" as the label.

**CKM\_TLS12\_MAC** This length should be in the range 0-20 (the output size of SHA-1 is 20 bytes). Signatures (MACs) produced by this mechanism will be taken from the start of the full 20-byte HMAC output.

Table xx, General-length TLS MAC: Key And Data Length

| **Function** | **Key type** | **Data length** | **Signature length** |
| --- | --- | --- | --- |
| C\_Sign | generic secret | any | 12 bytes |
| C\_Verify | generic secret | any | 12 bytes |

**CK\_TLS12\_MAC\_PARAMS** is a structure that provides the parameters to the **CKM\_TLS12\_MAC** mechanism. It is defined as follows:

typedef struct CK\_TLS12\_MAC\_PARAMS {

 CK\_MECHANISM\_TYPE prfHashFunction;

 CK\_ULONG ulServerOrClient;

} CK\_TLS12\_MAC\_PARAMS;

The fields of the structure have the following meanings:

 *prfHashFunction* the hash mechanism used in the TLS12 PRF construct. If the mechanism type CKM\_TLS\_PRF is used, this mechanism is should return the same data as one of the CKM\_TLS10\_\* mechanisms.

 *ulServerOrClient* 1 to use the label "server finished", 2 to use the label "client finished". All other values are invalid.

**CK\_TLS12\_MAC\_PARAMS\_PTR** is a pointer to a **CK\_TLS12\_MAC\_PARAMS**.

If accepted, **CKM\_TLS\_PRF** shall be deprecated/prohibited as a derivation mechanism. **CKM\_TLS\_MASTER\_KEY\_DERIVE** shall be deprecated, but permitted. **CKM\_TLS\_KEY\_AND\_MAC\_DERIVE** shall be deprecated, but permitted only if it does not allow the generation of "public" data (e.g. no data from the KDF shall be placed i