

EncryptionOptions.hpp

```
namespace verifone_sdk
```

```
enum class EncryptionOptions : unsigned
```

```
enumerator NONE
```

in because some options are incompatible.

```
enumerator DATA_PADDING_00
```

passed as hex strings: Data padding with 0x00: XX XX XX XX 00 00 00 00

```
enumerator DATA_PADDING_FF
```

< XX ff ff ff ff

```
enumerator TRUNCATE_PAN
```

789FFFF => 123456789F

```
enumerator ADD_EMPTY_TAGS
```

0) to result

```
enumerator TRUNCATE
```

is the fixed length

```
enumerator VALUE_ONLY
```

- Use the associated value only, without tag or length information. Not allowed if requesting more than one tag and one or more tags with.

```
enumerator TRACK2_EMV
```

- Format MSR Track 2 (TIV tag 57) according to the EMV Track 2 Equivalent Data spec. For CT and CTLS, this has no effect as they are V format. When not used with MSR, Track2 will contain start and end sentinels and LRC.

```
enumerator CVV_ASCII
```

- Format CVV as an ASCII string with no padding and no null terminator. When not used, CVV will be in BCD, with trailing F for odd

```
enumerator PAN_ASCII
```

- Format PAN as an ASCII string with no padding and no null terminator. When not used, PAN will be in BCD, with trailing F for odd

```
enumerator TRACK2_ASCII
```

- Format Track 2 as an ASCII string with no padding and no null terminator. When not used, Track2 will be in BCD, with trailing F for odd

```
enumerator EXPIRY_ASCII
```

<https://verifone.cloud/docs/in-person-payments/payment-software-development-kit-guide-psdk/api-reference/linux-api-40>

Updated: 17-Jul-2024

is an ASCII string with no padding

enumerator TRACK2_ASCII_MSR

ing according to MSR spec with start ';' and end '?' sentinels but with no LRC or padding

enumerator TRACK1_EMV

- Format Track1 equivalent data (TLV tag 56) according the EMV Track 1 Equivalent Data spec. For CT and CTLS, this has no effect as they are already in the expected EMV format. When not used with MSR, Track1 will contain start and end sentinels and LRC.

tions rhs) noexcept

constexpr std::string_view toString(const EncryptionOptions &arg)