```
                    <<interface>>
                        IMac
+MAC_KEY_MATERIAL: String = "gnu.crypto.mac.key.material"
+TRUNCATED_SIZE: String = "gnu.crypto.mac.truncated.size"
+name(): String
+macSize(): int
+init(attributes:Map): void
+update(b:byte): void
+update(in:byte[],offset:int,length:int): void
+digest(): byte[]
+reset(): void
+selfTest(): boolean
+clone(): Object
```

```
                  MacFactory
+getInstance(name:String): IMac
+getNames(): Set
```

```
                  HMacFactory
+getInstance(name:String): IMac
+getNames(): Set
```

```
                     BaseMac
#name: String
#underlyingHash: IMessageDigest
#truncatedSize: int
#<<create>> BaseMac(name:String)
#<<create>> BaseMac(name:String,underlyingHash:IMessageDigest)
+name(): String
+macSize(): int
+update(b:byte): void
+update(in:byte[],offset:int,length:int): void
+reset(): void
+clone(): Object
+init(attributes:Map): void
+digest(): byte[]
+selfTest(): boolean
```

```
                  MacInputStream
+<<create>> MacInputStream(in:InputStream,mac:IMac)
+getMac(): IMac
+setMac(mac:IMac): void
+on(flag:boolean): void
+read(): void
+read(buf:byte[],off:int,len:int): void
```

```
                  MacOutputStream
+<<create>> MacOutputStream(out:OutputStream,mac:IMac)
+getMac(): IMac
+setMac(mac:IMac): void
+on(flag:boolean): void
+write(b:byte): void
+write(buf:byte[],off:int,len:int): void
```

```
                     TMMH16
+TAG_LENGTH: String = "gnu.crypto.mac.tmmh.tag.length"
+KEYSTREAM: String = "gnu.crypto.mac.tmmh.keystream"
+PREFIX: String = "gnu.crypto.mac.tmmh.prefix"
```

```
   UHash32
```

```
                     UMac32
+NONCE_MATERIAL: String = "gnu.crypto.umac.nonce.material"
```

```
                      HMac
+USE_WITH_PKCS5_V2: String = "gnu.crypto.hmac.pkcs5"
#macSize: int
#blockSize: int
#ipadHash: IMessageDigest
#opadHash: IMessageDigest
#ipad: byte[]
#<<create>> HMac(underlyingHash:IMessageDigest)
+clone(): Object
+init(attributes:Map): void
+reset()
+digest(): byte[]
+selfTest(): boolean
```