

ClientFactory
<pre> +getInstance(name:String): ClientMechanism +getNames(): Set +createSaslClient(mechanisms:String[],authorizationID:String,protocol:String,serverName:String,props:Map,cbh:CallbackHandler): SaslClient </pre>

ServerFactory
<pre> +getInstance(name:String): ClientMechanism +getNames(): Set +createSaslServer(mechanism:String,protocol:String,serverName:String,props:Map,cbh:CallbackHandler): SaslClient </pre>

ClientMechanism
<pre> #mechanism: String #authorisationID: String #protocol: String #serverName: String #properties: Map #handler: CallbackHandler #complete: boolean = false #state: int = -1 #<<create>> ClientMechanism(name:String) #initMechanism(): void #resetMechanism(): void +evaluateChallenge(challenge:byte[]): byte[] +hasInitialResponse(): boolean +isComplete(): boolean +unwrap(in:byte[],offset:int,length:int): byte[] +wrap(out:byte[],offset:int,length:int): byte[] +getMechanismName(): String +getNegotiatedProperty(propName:String): Object +dispose(): void +getAuthorizationID(): String +getNegotiatedQOP(): String +getNegotiatedStrength(): String +getNegotiatedServerAuth(): String +getNegotiatedMaxBuffer(): String +getNegotiatedRawSendSize(): String +getNegotiatedPolicyNoPlainText(): String +getNegotiatedPolicyNoActive(): String +getNegotiatedPolicyNoDictionary(): String +getNegotiatedPolicyNoAnonymous(): String +getNegotiatedPolicyForwardSecrecy(): String +getNegotiatedPolicyPassCredentials(): String +getReuse(): String +engineUnwrap(in:byte[],offset:int,length:int): byte[] +engineWrap(out:byte[],offset:int,length:int): byte[] +init(attributes:Map): void +reset(): void </pre>

ServerMechanism
<pre> #mechanism: String #authorizationID: String #protocol: String #serverName: String #properties: Map #handler: CallbackHandler #complete: boolean = false #state: int = -1 #<<create>> ServerMechanism(name:String) #initMechanism(): void #resetMechanism(): void +evaluateResponse(response:byte[]): byte[] +isComplete(): boolean +unwrap(in:byte[],offset:int,length:int): byte[] +wrap(out:byte[],offset:int,length:int): byte[] +getMechanismName(): String +getNegotiatedProperty(propName:String): Object +dispose(): void +getAuthorizationID(): String +getNegotiatedQOP(): String +getNegotiatedStrength(): String +getNegotiatedServerAuth(): String +getNegotiatedMaxBuffer(): String +getNegotiatedRawSendSize(): String +getNegotiatedPolicyNoPlainText(): String +getNegotiatedPolicyNoActive(): String +getNegotiatedPolicyNoDictionary(): String +getNegotiatedPolicyNoAnonymous(): String +getNegotiatedPolicyForwardSecrecy(): String +getNegotiatedPolicyPassCredentials(): String +getReuse(): String +engineUnwrap(in:byte[],offset:int,length:int): byte[] +engineWrap(out:byte[],offset:int,length:int): byte[] +init(attributes:Map): void +reset(): void </pre>

<<interface>> IAuthInfoProviderFactory
<pre> +getInstance(mechanism:String): IAuthInfoProvider </pre>

<<interface>> IAuthInfoProvider
<pre> +activate(context:Map): void +passivate(): void +contains(userName:String): boolean +lookup(userID:Map): Map +update(userCredentials:Map): void +getConfiguration(mode:String): Map </pre>

AuthInfoProviderFactory
<pre> +getInstance(mechanism:String): IAuthInfoProvider </pre>
AuthInfo
<pre> +getProvider(mechanism:String): IAuthInfoProvider </pre>

SaslOutputStream
<pre> +<<create>> SaslOutputStream(client:SaslClient,dest:OutputStream) +<<create>> SaslOutputStream(server:SaslServer,dest:OutputStream) +close(): void +flush(): void +write(b:int): void +write(b:byte[],off:int,len:int): void </pre>

SaslInputStream
<pre> +<<create>> SaslInputStream(client:SaslClient,source:InputStream) +<<create>> SaslInputStream(server:SaslServer,source:InputStream) +available(): int +close(): void +read(): int +read(b:byte[],off:int,len:int): int </pre>

InputBuffer
<pre> #in: ByteArrayInputStream #length: int +<<create>> InputBuffer(frame:byte[]) +getInstance(raw:byte[]): InputBuffer +getInstance(raw:byte[],offset:int,length:int): InputBuffer +fourBytesToLength(b:byte[]): int +twoBytesToLength(b:byte[]): int +hasMoreElements(): boolean +getScalar(count:int): long +getOS(): byte[] +getEOS(): byte[] +getMPI(): BigInteger +getText(): String </pre>

OutputBuffer
<pre> +<<create>> OutputBuffer() +setScalar(count:int,b:int): void +setOS(b:byte[]): void +setEOS(b:byte[]): void +setMPI(val:BigInteger): void +setText(str:String): void +encode(): byte[] +wrap(): byte[] </pre>

IntegrityException

ConfidentialityException

IllegalMechanismStateException

UserAlreadyExistsException

SaslEncodingException

NoSuchMechanismException

NoSuchUserException