

# Amazon S3 Compatible API

v1.0.0 KSAN이 지원하는 Amazon S3 Compatible API에 대한 정보를 제공합니다.

- [AbortMultipartUpload](#)
- [CompleteMultipartUpload](#)
- [CopyObject](#)
- [CreateBucket](#)
- [CreateMultipartUpload](#)
- [DeleteBucket](#)
- [DeleteBucketCors](#)
- [DeleteBucketEncryption](#)
- [DeleteBucketLifecycle](#)
- [DeleteBucketPolicy](#)
- [DeleteBucketReplication](#)
- [DeleteBucketTagging](#)
- [DeleteBucketWebsite](#)
- [DeleteObject](#)
- [DeleteObjects](#)
- [DeleteObjectTagging](#)
- [DeletePublicAccessBlock](#)
- [GetBucketAcl](#)
- [GetBucketCors](#)
- [GetBucketEncryption](#)
- [GetBucketLifecycleConfiguration](#)
- [GetBucketLocation](#)
- [GetBucketPolicy](#)
- [GetBucketPolicyStatus](#)
- [GetBucketReplication](#)
- [GetBucketTagging](#)
- [GetBucketVersioning](#)
- [GetBucketWebsite](#)
- [GetObject](#)
- [GetObjectAcl](#)
- [GetObjectRetention](#)
- [GetObjectTagging](#)
- [GetPublicAccessBlock](#)
- [HeadBucket](#)
- [HeadObject](#)
- [ListBuckets](#)
- [ListMultipartUploads](#)
- [ListObjects](#)
- [ListObjectsV2](#)
- [ListObjectVersions](#)
- [ListParts](#)
- [PutBucketAcl](#)
- [PutBucketCors](#)
- [PutBucketEncryption](#)
- [PutBucketLifecycleConfiguration](#)
- [PutBucketPolicy](#)
- [PutBucketReplication](#)
- [PutBucketTagging](#)
- [PutBucketVersioning](#)

- [PutBucketWebsite](#)
- [PutObject](#)
- [PutObjectAcl](#)
- [PutObjectLockConfiguration](#)
- [PutObjectTagging](#)
- [PutPublicAccessBlock](#)
- [UploadPart](#)
- [UploadPartCopy](#)

# AbortMultipartUpload

## Overview

Multipart Upload를 중단합니다. 중단 된 이후에는 uploadID를 사용하여 UploadPart를 요청 할 수 없습니다.

## Request Syntax

```
DELETE /Bucket/Key?uploadId=UploadId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /bucket/example-object?uploadId=RxnIeEPx3G6n61T HTTP/1.1
Host: 192.168.111.111:8080
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	Multipart Upload가 수행되어지는 Bucket 명	Yes
Key	Multipart Upload가 수행되어지는 Object 명	Yes
uploadId	Multipart Upload를 식별하는 ID	Yes
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	
x-amz-expected-bucket-owner	예상되는 bucket 소유자의 ID. bucket이 다른 계정의 소유자인 경우 HTTP 403 (Access Denied)으로 요청 이 실패합니다.	

## Request Body

요청 본문이 없습니다.

## Response Syntax

```
HTTP/1.1 204
x-amz-request-charged: RequestCharged
```

## Response Elements

Parameter Name	Description	Required
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	

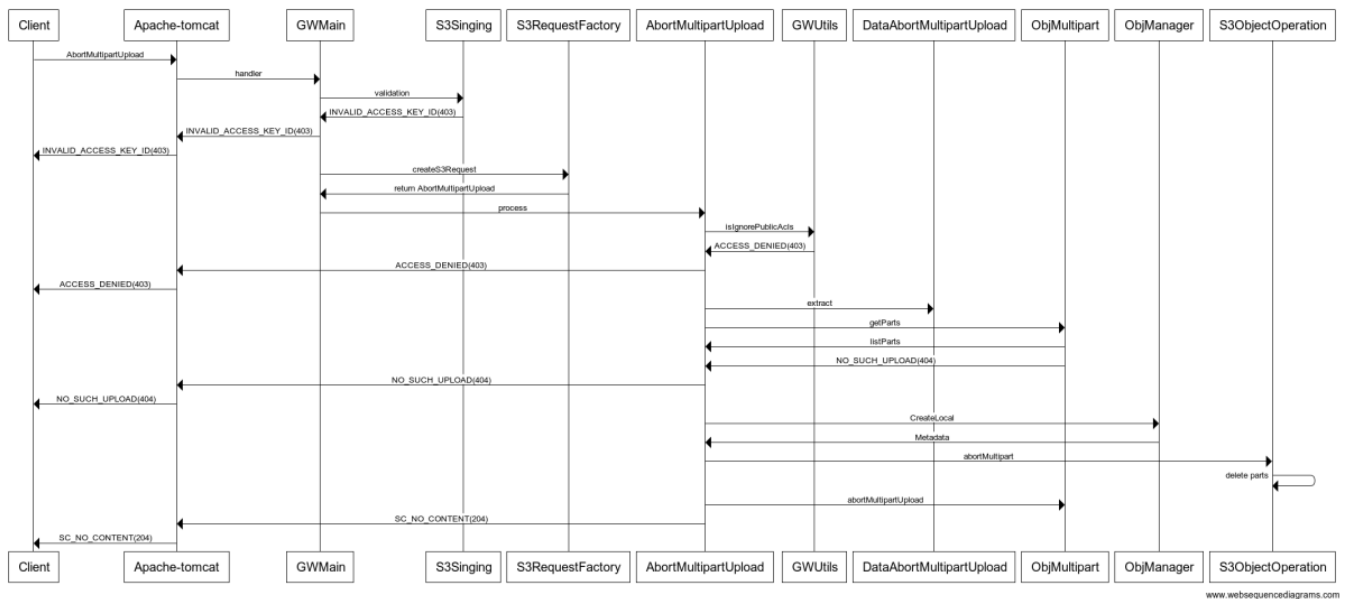
# Sample Response

```
HTTP/1.1 204 OK
x-amz-id-2: Weag1LuByRx9e6j50nimru9p04ZVKnJ2Qz7/C1NPcfTWAtRPfTa0Fg==
x-amz-request-id: 996c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
Content-Length: 0
```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Invalid access key)	access key가 유효하지 않음
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Such uploadId)	uploadId가 존재하지 않음
500 (Server Error)	

## Processing of requests



# CompleteMultipartUpload

## Overview

이전에 업로드한 Parts를 지정한 Key로 합쳐서 Multipart Upload를 완료합니다.

## Request Syntax

```
POST /Bucket/Key?uploadId=UploadId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<CompleteMultipartUpload xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Part>
    <ETag>string</ETag>
    <PartNumber>integer</PartNumber>
  </Part>
  ...
</CompleteMultipartUpload>
```

## Sample Request

```
POST /example-bucket/example-object?uploadId=AAAsb2FkIElEIGZvcjBlbHZpbmcncyWeeS1tb3ZpZS5tMnRzIRRwbG9hZA HTTP/1.1
Host: 192.168.111.111:8080
Date: Mon, 1 Nov 2010 20:34:56 GMT
Content-Length: 391
Authorization: authorization string

<CompleteMultipartUpload>
  <Part>
    <PartNumber>1</PartNumber>
    <ETag>"a54357aff0632cce46d942af68356b38"</ETag>
  </Part>
  <Part>
    <PartNumber>2</PartNumber>
    <ETag>"0c78aef83f66abc1fa1e8477f296d394"</ETag>
  </Part>
  <Part>
    <PartNumber>3</PartNumber>
    <ETag>"acbd18db4cc2f85cedef654fccc4a4d8"</ETag>
  </Part>
</CompleteMultipartUpload>
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	Multipart Upload가 수행되어지는 Bucket 명	Yes
Key	Multipart Upload가 수행되어지는 Object 명	Yes
uploadId	Multipart Upload를 식별하는 ID	Yes
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	
x-amz-expected-bucket-owner	예상되는 bucket 소유자의 ID. bucket이 다른 계정의 소유자인 경우 HTTP 403 (Access Denied)으로 요청이 실패합니다.	

# Request Body

## CompleteMultipartUpload

```
<CompleteMultipartUpload xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Part>
    <ETag>string</ETag>
    <PartNumber>integer</PartNumber>
  </Part>
  ...
</CompleteMultipartUpload>
```

# Response Syntax

```
HTTP/1.1 200
x-amz-expiration: Expiration
x-amz-version-id: VersionId
<?xml version="1.0" encoding="UTF-8"?>
<CompleteMultipartUploadResult>
  <Location>string</Location>
  <Bucket>string</Bucket>
  <Key>string</Key>
  <ETag>string</ETag>
</CompleteMultipartUploadResult>
```

# Response Elements

Parameter Name	Description	Required
x-amz-expiration	개체 만료가 구성된 경우 응답에 이 헤더가 포함됩니다.	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
x-amz-server-side-encryption	Amazon S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘 Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	있는 경우 개체에 사용된 AWS KMS(AWS Key Management Service) 대칭 고객 관리 키의 ID를 지정합니다.	
x-amz-server-side-encryption-bucket-key-enabled	복사된 개체가 AWS KMS(SSE-KMS)와의 서버측 암호화에 S3 버킷 키를 사용하는지 여부를 나타냅니다.	
x-amz-version-id	새로 만든 복사본의 버전 ID입니다.	
CompleteMultipartUploadResult	Response Syntax 참조	

# Sample Response

```
HTTP/1.1 200 OK
x-amz-id-2: Uuag1LuByRx9e6j50nimru9p04ZVKnJ2Qz7/C1NPcfTWAtrPfTa0Fg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
Connection: close
Server: AmazonS3

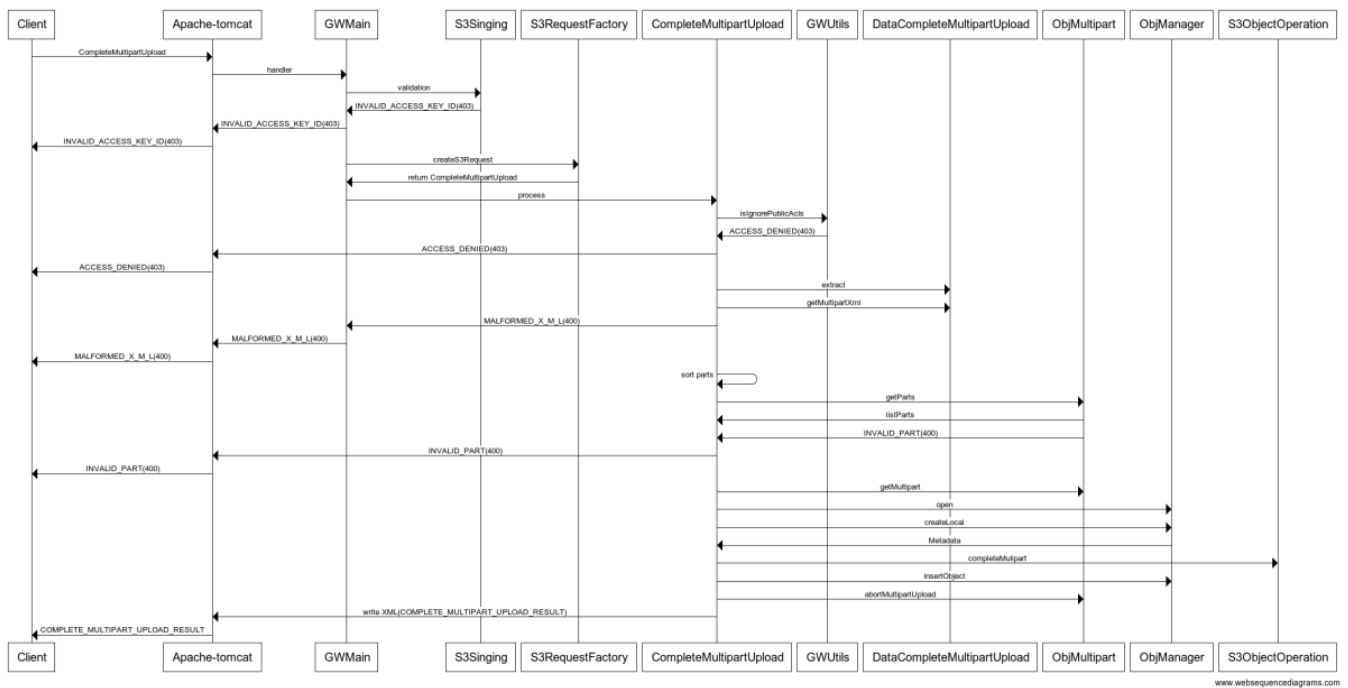
<?xml version="1.0" encoding="UTF-8"?>
  <CompleteMultipartUploadResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
    <Location>http://Example-Bucket.s3.amazonaws.com/Example-Object</Location>
    <Bucket>Example-Bucket</Bucket>
    <Key>Example-Object</Key>
```

```
<ETag>"3858f62230ac3c915f300c664312c11f-9"</ETag>
</CompleteMultipartUploadResult>
```

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
400 (MALFORMED_X_M_L)	XML 형식이 유효하지 않음
400 (INVALID_PART)	Part를 찾을 수 없음
403 (Invalid access key)	access key가 유효하지 않음
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Such uploadId)	uploadId가 존재하지 않음
500 (Server Error)	

# Processing of requests



# CopyObject

## Overview

S3에 이미 저장된 개체의 복사본을 생성합니다.

## Request Syntax

```
PUT /Bucket/Key HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-acl: ACL
Cache-Control: CacheControl
Content-Disposition: ContentDisposition
Content-Encoding: ContentEncoding
Content-Language: ContentLanguage
Content-Type: ContentType
x-amz-copy-source: CopySource
x-amz-copy-source-if-match: CopySourceIfMatch
x-amz-copy-source-if-modified-since: CopySourceIfModifiedSince
x-amz-copy-source-if-none-match: CopySourceIfNoneMatch
x-amz-copy-source-if-unmodified-since: CopySourceIfUnmodifiedSince
Expires: Expires
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write-acp: GrantWriteACP
x-amz-metadata-directive: MetadataDirective
x-amz-tagging-directive: TaggingDirective
x-amz-server-side-encryption: ServerSideEncryption
x-amz-storage-class: StorageClass
x-amz-website-redirect-location: WebsiteRedirectLocation
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-context: SSEKMSEncryptionContext
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-copy-source-server-side-encryption-customer-algorithm: CopySourceSSECustomerAlgorithm
x-amz-copy-source-server-side-encryption-customer-key: CopySourceSSECustomerKey
x-amz-copy-source-server-side-encryption-customer-key-MD5: CopySourceSSECustomerKeyMD5
x-amz-request-payer: RequestPayer
x-amz-tagging: Tagging
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
x-amz-object-lock-legal-hold: ObjectLockLegalHoldStatus
x-amz-expected-bucket-owner: ExpectedBucketOwner
x-amz-source-expected-bucket-owner: ExpectedSourceBucketOwner
```

## Sample Request

```
PUT /bucket/my-second-image.jpg HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 28 Oct 2009 22:32:00 GMT
x-amz-copy-source: /bucket/my-image.jpg?
versionId=3/L4kqtJlcpXroTDmJ+rmSpXd3dIbrHY+MTRCxf3vjVBH40Nr8X8gdRQBpUMLUo
Authorization: authorization string
```



# URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
Cache-Control	요청/응답 체인을 따라 캐싱 동작을 지정합니다.	
Content-Disposition	개체에 대한 프레젠테이션 정보를 지정합니다.	
Content-Encoding	개체에 적용된 콘텐츠 인코딩과 콘텐츠 유형 헤더 필드가 참조하는 미디어 유형을 가져오기 위해 적용해야 하는 디코딩 메커니즘을 지정합니다.	
Content-Language	콘텐츠가 사용되는 언어입니다.	
Content-Type	내용 유형	
Expires	개체를 더 이상 캐시할 수 없는 날짜 및 시간입니다.	
x-amz-acl	개체에 적용할 미리 제공된(canned) ACL.	
x-amz-copy-source	복사 작업의 원본 개체를 지정합니다.	
x-amz-copy-source-if-match	해당 엔티티 태그(ETAG)가 지정된 태그와 일치하는 경우 개체를 복사합니다.	
x-amz-copy-source-if-modified-since	지정된 시간 이후 수정한 경우 개체를 복사합니다.	
x-amz-copy-source-if-none-match	해당 엔티티 태그(ETAG)가 지정한 ETAG와 다른 경우 개체를 복사합니다.	
x-amz-copy-source-if-unmodified-since	지정한 시간 이후 개체를 수정하지 않은 경우 개체를 복사합니다.	
x-amz-copy-source-server-side-encryption-customer-algorithm	원본 개체의 암호를 해독할 때 사용할 알고리즘(예: AES256)을 지정합니다.	
x-amz-copy-source-server-side-encryption-customer-key	소스 개체의 암호를 해독하는 데 사용할 Amazon S3에 대해 고객이 제공한 암호화 키를 지정합니다. 이 헤더에 제공된 암호화 키는 원본 개체를 만들 때 사용된 암호화 키여야 합니다.	
x-amz-copy-source-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 128비트 MD5 다이제스트를 지정합니다. Amazon S3는 메시지 무결성 검사에 이 헤더를 사용하여 암호화 키가 오류 없이 전송되었는지 확인합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-grant-full-control	허가자에게 개체에 대한 READ, READ_ACL, WRITE_ACP 권한을 부여합니다.	
x-amz-grant-read	부여자가 개체 데이터와 해당 메타데이터를 읽을 수 있습니다.	
x-amz-grant-read-acp	허가자가 개체 ACL을 읽을 수 있습니다.	
x-amz-grant-write-acp	허가자가 해당 개체에 대한 ACL을 쓸 수 있습니다.	
x-amz-metadata-directive	메타데이터를 원본 개체에서 복사할지 아니면 요청에 제공된 메타데이터로 바꿀지를 지정합니다. Values : COPY or REPLACE	
x-amz-object-lock-legal-hold	복사된 개체에 증거 보존을 적용할지 여부를 지정합니다. Values : ON or OFF	
x-amz-object-lock-mode	복사된 개체에 적용할 개체 Lock 모드입니다.	
x-amz-object-lock-retain-until-date	복사된 개체의 개체 잠금이 만료되는 날짜 및 시간입니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	
x-amz-server-side-encryption	S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘(예: AES256, aws:kms) Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	개체 암호화에 사용할 AWS KMS 키 ID를 지정합니다. SSL을 통하거나 SigV4를 사용하지 않으면 AWS KMS로 보호되는 개체에 대한 모든 GET 및 PUT 요청이 실패합니다.	
x-amz-server-side-encryption-bucket-key-enabled	AWS KMS(SSE-KMS)를 사용한 서버측 암호화의 개체 암호화에 Amazon S3 버킷 키를 사용해야 하는지 여부를 지정합니다.	

Parameter Name	Description	Required
x-amz-server-side-encryption-context	개체 암호화에 사용할 AWS KMS 암호화 컨텍스트를 지정합니다.	
x-amz-server-side-encryption-customer-algorithm	개체를 암호화할 때 사용할 알고리즘(예: AES256)을 지정합니다.	
x-amz-server-side-encryption-customer-key	데이터 암호화에 사용할 S3에 대해 고객이 제공한 암호화 키를 지정합니다.	
x-amz-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 128비트 MD5 다이제스트를 지정합니다.	
x-amz-source-expected-bucket-owner	예상 소스 버킷 소유자의 계정 ID입니다.	
x-amz-storage-class	기본적으로 S3는 STANDARD 스토리지 클래스를 사용하여 새로 생성된 개체를 저장합니다.	
x-amz-tagging	개체 대상 개체의 태그 집합은 TaggingDirective와 함께 사용해야 합니다. 태그 집합은 URL 쿼리 매개 변수로 인코딩되어야 합니다.	
x-amz-tagging-directive	개체 태그 집합을 원본 개체에서 복사할지 또는 요청에 제공된 태그 집합으로 바꿀지를 지정합니다. Value : COPY or REPLACE	
x-amz-website-redirect-location	버킷이 웹 사이트로 구성된 경우 이 개체에 대한 요청을 동일한 버킷의 다른 개체 또는 외부 URL로 리디렉션합니다.	

## Request Body

## Response Syntax


```

HTTP/1.1 200
x-amz-expiration: Expiration
x-amz-copy-source-version-id: CopySourceVersionId
x-amz-version-id: VersionId
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-context: SSEKMSEncryptionContext
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<CopyObjectResult>
  <ETag>string</ETag>
  <LastModified>timestamp</LastModified>
</CopyObjectResult>

```

## Response Elements

Parameter Name	Description	Required
x-amz-copy-source-version-id	대상 버킷에 복사된 개체의 버전입니다.	
x-amz-expiration	개체 만료가 구성된 경우 응답에 이 헤더가 포함됩니다.	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
x-amz-server-side-encryption	Amazon S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘 Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	있는 경우 개체에 사용된 AWS KMS(AWS Key Management Service) 대칭 고객 관리 키의 ID를 지정합니다.	

Parameter Name	Description	Required
x-amz-server-side-encryption-bucket-key-enabled	복사된 개체가 AWS KMS(SSE-KMS)와의 서버측 암호화에 S3 버킷 키를 사용하는지 여부를 나타냅니다.	
x-amz-server-side-encryption-context	존재하는 경우 개체 암호화에 사용할 AWS KMS 암호화 컨텍스트를 지정합니다.	
x-amz-server-side-encryption-customer-algorithm	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 사용된 암호화 알고리즘을 확인하는 헤더가 포함됩니다.	
x-amz-server-side-encryption-customer-key-MD5	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 이 헤더가 포함되어 고객이 제공한 암호화 키에 대한 왕복 메시지 무결성 확인을 제공합니다. 	
x-amz-version-id	새로 만든 복사본의 버전 ID입니다.	
CopyObjectResult	Response Syntax 참조	Yes

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TnqcoF8eFidJG9Z/2mkiDFu8yU9AS1ed40pIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
x-amz-version-id: QUpfndnhfd8438MNFNDN93jdnJFkdmqnh893
x-amz-copy-source-version-id: 09df8234529fjs0dfi0w52935029wefdj
Date: Wed, 28 Oct 2009 22:32:00 GMT
Connection: close
Server: AmazonS3

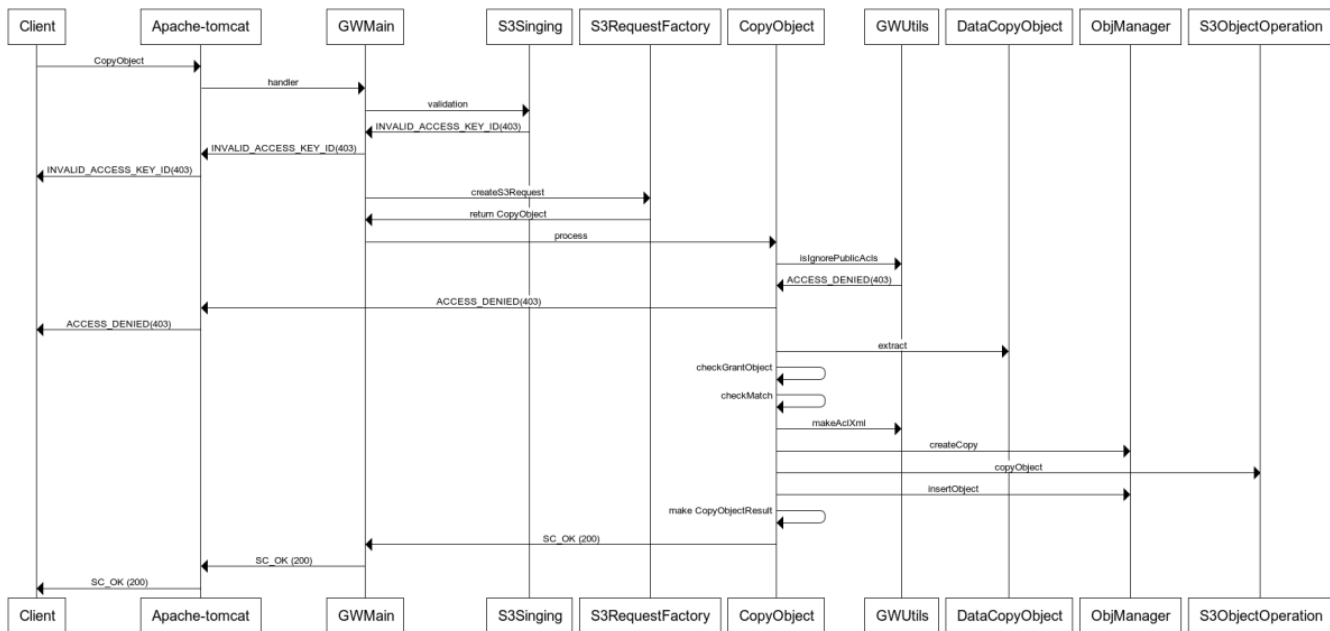
<?xml version="1.0" encoding="UTF-8"?>
<CopyObjectResult>
  <LastModified>2009-10-28T22:32:00</LastModified>
  <ETag>"9b2cf535f27731c974343645a3985328"</ETag>
</CopyObjectResult>

```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Such uploadId)	uploadId가 존재하지 않음
500 (Server Error)	

## Processing of requests



# CreateBucket

## Overview

새 S3 버킷을 만듭니다. 버킷을 생성하려면 S3에 등록하고 요청을 인증할 수 있는 유효한 AWS 액세스 키 ID가 있어야 합니다. 익명 요청은 버킷을 만들 수 없습니다. 버킷을 생성하면 버킷 소유자가 됩니다.

## Request Syntax

```
PUT /Bucket HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-acl: ACL
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write: GrantWrite
x-amz-grant-write-acp: GrantWriteACP
x-amz-bucket-object-lock-enabled: ObjectLockEnabledForBucket
x-amz-object-ownership: ObjectOwnership
<?xml version="1.0" encoding="UTF-8"?>
<CreateBucketConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <LocationConstraint>string</LocationConstraint>
</CreateBucketConfiguration>
```

## Sample Request

```
PUT /Bucket/ httpHTTP/1.1
Host: 192.168.111.111:8080
Content-Length: 0
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
```

## URI Request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-acl	개체에 적용할 미리 제공된(canned) ACL. Values : private or public-read or public-read-write or authenticated-read	
x-amz-bucket-object-lock-enabled	새 버킷에 대해 S3 Object Lock을 사용할지 여부를 지정합니다.	
x-amz-grant-full-control	허가자에게 개체에 대한 READ, READ_ACL, WRITE_ACP 권한을 부여합니다.	
x-amz-grant-read	부여자가 개체 데이터와 해당 메타데이터를 읽을 수 있습니다.	
x-amz-grant-read-acp	허가자가 개체 ACL을 읽을 수 있습니다.	
x-amz-grant-write	허가자가 버킷에 새 개체를 만들 수 있습니다.	
x-amz-grant-write-acp	허가자가 해당 개체에 대한 ACL을 쓸 수 있습니다.	
x-amz-object-ownership	버킷의 소유권 컨트롤에 대한 개체 소유권을 위한 컨테이너 요소입니다.	

## Request Body

```
<CreateBucketConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <LocationConstraint>string</LocationConstraint>
</CreateBucketConfiguration>
```

# Response Syntax

```
HTTP/1.1 200
Location: Location
```

# Response Elements

Parameter Name	Description	Required
Location	버킷을 만들 영역을 지정합니다.	

# Sample Response

```
HTTP/1.1 200 OK
```

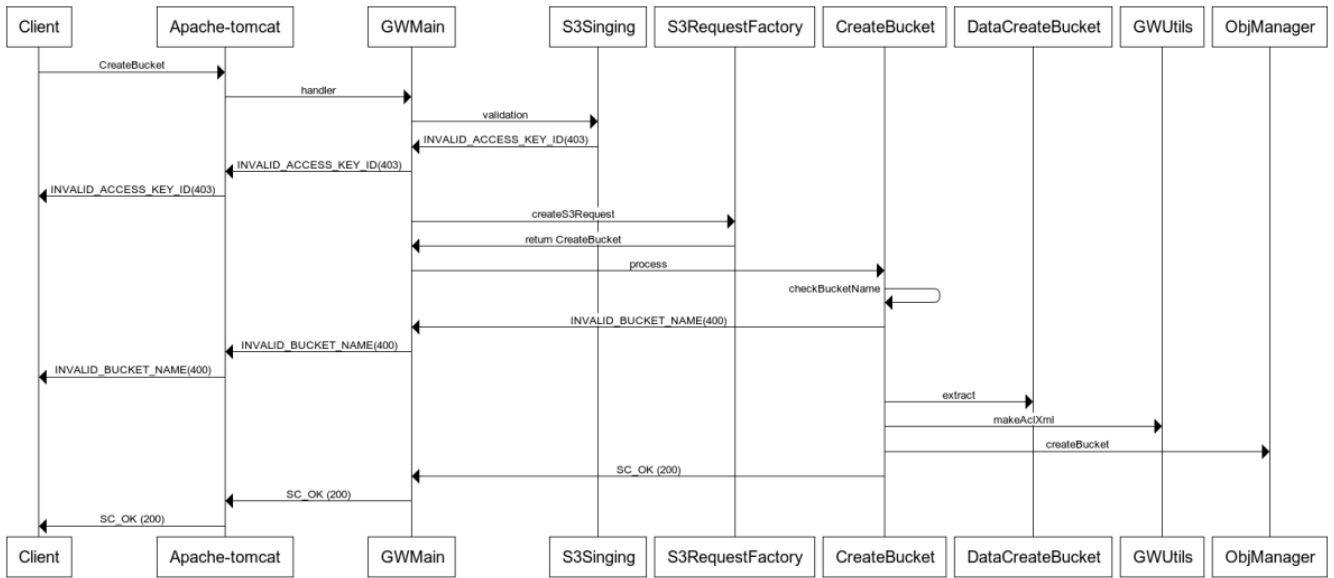
```
HTTP/1.1 200 OK
x-amz-id-2: YgIPiFbiKa2bj0KMg95r/0zo3emzU4dzsD4rcKCHQUAdQkf3ShJT00pXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Wed, 01 Mar 2006 12:00:00 GMT

Location: /colorpictures
Content-Length: 0
Connection: close
Server: KSAN-GW
```

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Invalid Bucket Name)	버킷명이 유효하지 않음
403 (Access Denied)	권한 없음
405 (Method Not Allowed)	버킷명이 null 건내 짐
409 (Bucket Already Owned By You)	버킷이 이미 존재하고 요청자의 소유임
409 (Bucket Already Exists)	버킷이 이미 존재 함
500 (Server Error)	

# Processing of requests



# CreateMultipartUpload

## Overview

Multipart Upload를 시작하고 upload ID를 반환합니다.

## Request Syntax

```
POST /Bucket/Key?uploads HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-acl: ACL
Cache-Control: CacheControl
Content-Disposition: ContentDisposition
Content-Encoding: ContentEncoding
Content-Language: ContentLanguage
Content-Type: ContentType
Expires: Expires
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write-acp: GrantWriteACP
x-amz-server-side-encryption: ServerSideEncryption
x-amz-storage-class: StorageClass
x-amz-website-redirect-location: WebsiteRedirectLocation
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-context: SSEKMSEncryptionContext
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-request-payer: RequestPayer
x-amz-tagging: Tagging
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
x-amz-object-lock-legal-hold: ObjectLockLegalHoldStatus
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
POST /ExampleBucket/example-object?uploads HTTP/1.1
Host: 192.168.111.111:8080
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string
```

## URI Request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
Cache-Control	요청/응답 체인을 따라 캐싱 동작을 지정합니다.	
Content-Disposition	개체에 대한 프레젠테이션 정보를 지정합니다.	
Content-Encoding	개체에 적용된 콘텐츠 인코딩과 콘텐츠 유형 헤더 필드가 참조하는 미디어 유형을 가져오기 위해 적용해야 하는 디코딩 메커니즘을 지정합니다.	
Content-Language	콘텐츠가 사용되는 언어입니다.	



Parameter Name	Description	Required
Content-Type	내용 유형	
Expires	개체를 더 이상 캐시할 수 없는 날짜 및 시간입니다.	
x-amz-acl	개체에 적용할 미리 제공된(canned) ACL.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-grant-full-control	허가자에게 개체에 대한 READ, READ_ACL, WRITE_ACP 권한을 부여합니다.	
x-amz-grant-read	부여자가 개체 데이터와 해당 메타데이터를 읽을 수 있습니다.	
x-amz-grant-read-acp	허가자가 개체 ACL을 읽을 수 있습니다.	
x-amz-grant-write	허가자가 버킷에 새 개체를 만들 수 있습니다.	
x-amz-grant-write-acp	허가자가 해당 개체에 대한 ACL을 쓸 수 있습니다.	
x-amz-metadata-directive	메타데이터를 원본 개체에서 복사할지 아니면 요청에 제공된 메타데이터로 바꿀지를 지정합니다. Values : COPY or REPLACE	
x-amz-object-lock-legal-hold	복사된 개체에 증거 보존을 적용할지 여부를 지정합니다. Values : ON or OFF	
x-amz-object-lock-mode	복사된 개체에 적용할 개체 Lock 모드입니다.	
x-amz-object-lock-retain-until-date	복사된 개체의 개체 잠금이 만료되는 날짜 및 시간입니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	
x-amz-server-side-encryption	S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘(예: AES256, aws:kms) Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	개체 암호화에 사용할 AWS KMS 키 ID를 지정합니다. SSL을 통하거나 SigV4를 사용하지 않으면 AWS KMS로 보호되는 개체에 대한 모든 GET 및 PUT 요청이 실패합니다. 	
x-amz-server-side-encryption-bucket-key-enabled	AWS KMS(SSE-KMS)를 사용한 서버측 암호화의 개체 암호화에 Amazon S3 버킷 키를 사용해야 하는지 여부를 지정합니다.	
x-amz-server-side-encryption-context	개체 암호화에 사용할 AWS KMS 암호화 컨텍스트를 지정합니다.	
x-amz-server-side-encryption-customer-algorithm	개체를 암호화할 때 사용할 알고리즘(예: AES256)을 지정합니다.	
x-amz-server-side-encryption-customer-key	데이터 암호화에 사용할 S3에 대해 고객이 제공한 암호화 키를 지정합니다. 	
x-amz-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 128비트 MD5 다이제스트를 지정합니다. 	
x-amz-source-expected-bucket-owner	예상 소스 버킷 소유자의 계정 ID입니다.	
x-amz-storage-class	기본적으로 S3는 STANDARD 스토리지 클래스를 사용하여 새로 생성된 개체를 저장합니다.	
x-amz-tagging	개체 대상 개체의 태그 집합은 TaggingDirective와 함께 사용해야 합니다. 태그 집합은 URL 쿼리 매개 변수로 인코딩되어야 합니다.	
x-amz-tagging-directive	개체 태그 집합을 원본 개체에서 복사할지 또는 요청에 제공된 태그 집합으로 바꿀지를 지정합니다. Value : COPY or REPLACE 	
x-amz-website-redirect-location	버킷이 웹 사이트로 구성된 경우 이 개체에 대한 요청을 동일한 버킷의 다른 개체 또는 외부 URL로 리디렉션합니다.	

## Request Body

## Response Syntax

```

x-amz-abort-date: AbortDate
x-amz-abort-rule-id: AbortRuleId
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-context: SSEKMSEncryptionContext
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<InitiateMultipartUploadResult>
  <Bucket>string</Bucket>
  <Key>string</Key>
  <UploadId>string</UploadId>
</InitiateMultipartUploadResult>

```

## Response Elements

Parameter Name	Description	Required
x-amz-abort-date	버킷에 불완전한 다중 파트 업로드를 중단하는 작업으로 구성된 수명 주기 규칙이 있고 수명 주기 규칙의 접두사가 요청의 개체 이름과 일치하면 시작된 다중 파트 업로드를 중단 작업에 적합한 시기를 나타내는 이 헤더가 응답에 포함됩니다.	
x-amz-abort-rule-id	x-amz-abort-date 헤더와 함께 반환됩니다. 불완전한 다중 파트 업로드를 중단하는 작업을 정의하는 적용 가능한 라이프사이클 구성 규칙을 식별합니다.	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
x-amz-server-side-encryption-aws-kms-key-id	있는 경우 개체에 사용된 AWS KMS(AWS Key Management Service) 대칭 고객 관리 키의 ID를 지정합니다.	
x-amz-server-side-encryption-bucket-key-enabled	복사된 개체가 AWS KMS(SSE-KMS)와의 서버측 암호화에 S3 버킷 키를 사용하는지 여부를 나타냅니다.	
x-amz-server-side-encryption-context	존재하는 경우 개체 암호화에 사용할 AWS KMS 암호화 컨텍스트를 지정합니다.	
x-amz-server-side-encryption-customer-algorithm	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 사용된 암호화 알고리즘을 확인하는 헤더가 포함됩니다.	
x-amz-server-side-encryption-customer-key-MD5	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 이 헤더가 포함되어 고객이 제공한 암호화 키에 대한 왕복 메시지 무결성 확인을 제공합니다.	
InitiateMultipartUploadResult	Response Syntax 참조	

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: Uuag1LuByRx9e6j50nimru9p04ZVKnJ2Qz7/C1NPcfTWAtRPfTa0Fg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
Transfer-Encoding: chunked
Connection: keep-alive
Server: KSAN-GW

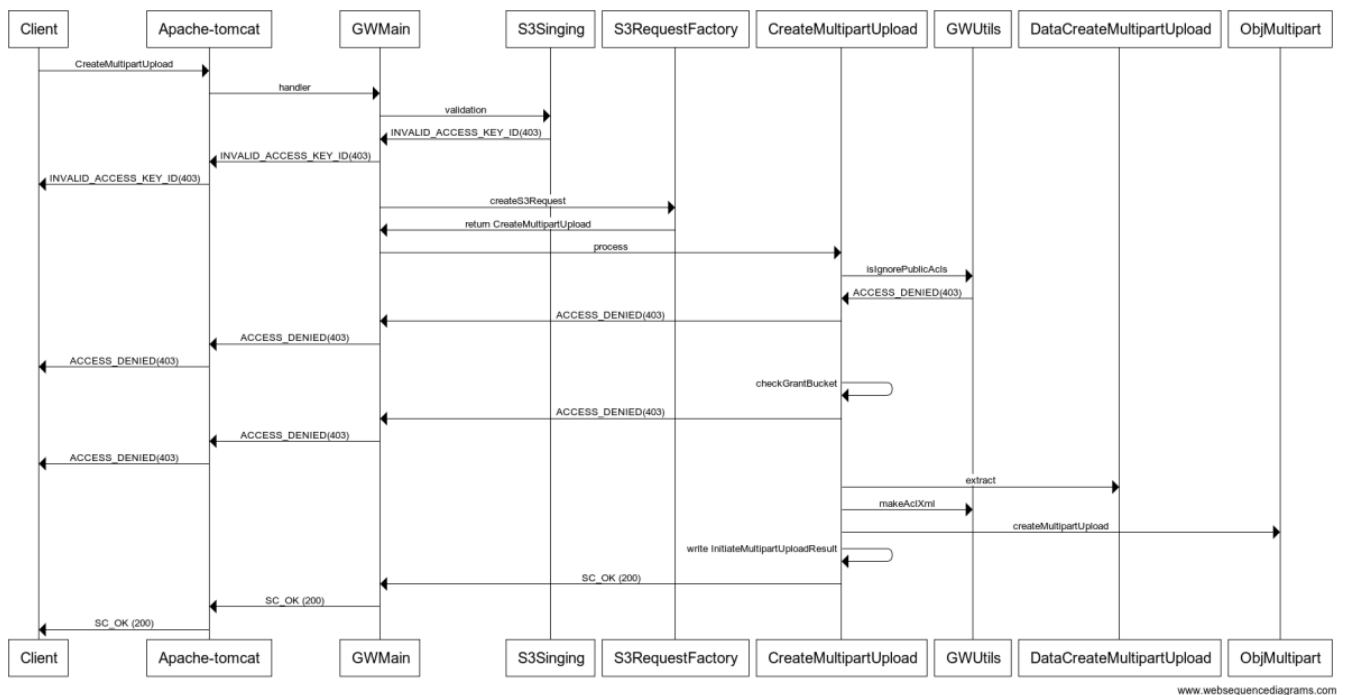
<?xml version="1.0" encoding="UTF-8"?>
<InitiateMultipartUploadResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Bucket>example-bucket</Bucket>
  <Key>example-object</Key>
  <UploadId>VXBsb2FkIElEIGZvcjA2aWwpcmcncyBteS1tb3ZpZS5tMnRzIHVwbG9hZA</UploadId>
</InitiateMultipartUploadResult>

```

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
400 (MALFORMED_X_M_L)	XML 형식이 유효하지 않음
400 (INVALID_PART)	Part를 찾을 수 없음
403 (Invalid access key)	access key가 유효하지 않음
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Such uploadId)	uploadId가 존재하지 않음
500 (Server Error)	

## Processing of requests



# DeleteBucket

## Overview

S3 버킷을 삭제합니다.

## Request Syntax

```
DELETE /Bucket/ HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /quotes/ HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 01 Mar 2006 12:00:00 GMT
Authorization: authorization string
```

## URI Request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

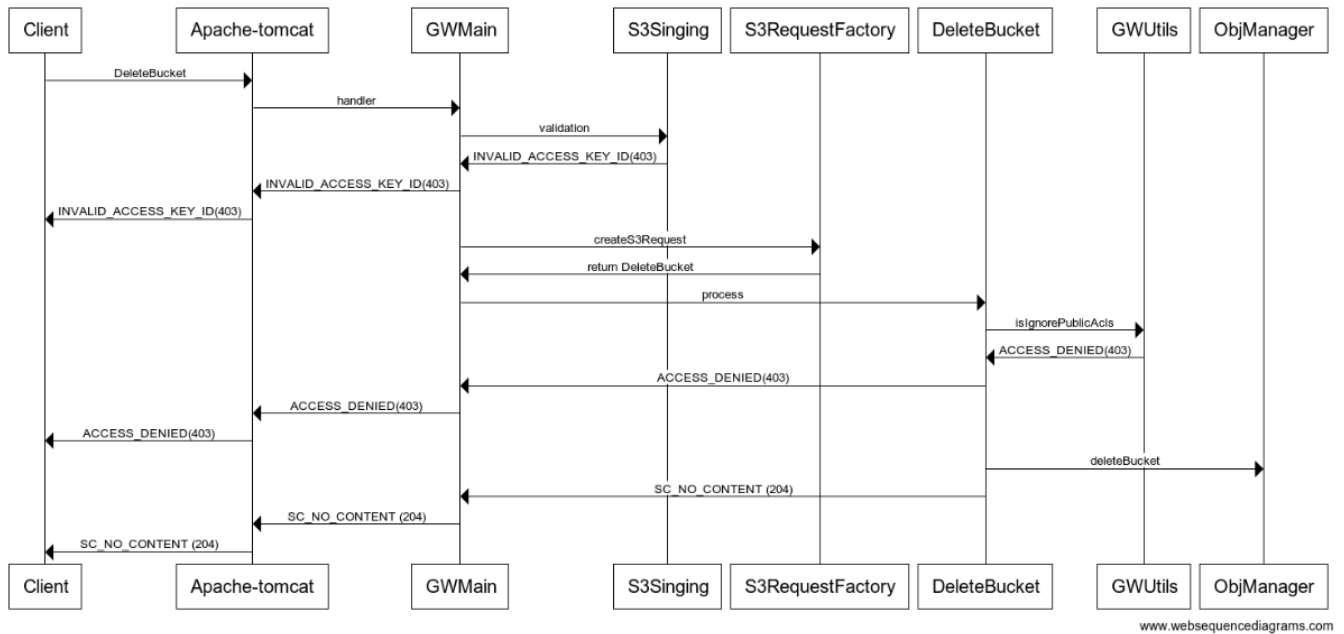
## Sample Response

```
HTTP/1.1 204 No Content
x-amz-id-2: JuKZqmXuiwFeDQxhD7M8KtsKobSzWA1QEjLbTMTagkKdBX2z7Il/jGhDeJ3j6s80
x-amz-request-id: 32FE2CEB32F5EE25
Date: Wed, 01 Mar 2006 12:00:00 GMT
Connection: close
Server: KSAN-GW
```

# HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# DeleteBucketCors

## Overview

버킷에 대한 CORS(Cross-Origin Resource Sharing) 정보 집합을 삭제합니다.

## Request Syntax

```
DELETE /Bucket/?cors HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /ExampleBucket/?cors HTTP/1.1
Host: 192.168.111.111:8080
Date: Tue, 13 Dec 2011 19:14:42 GMT
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

## Sample Response

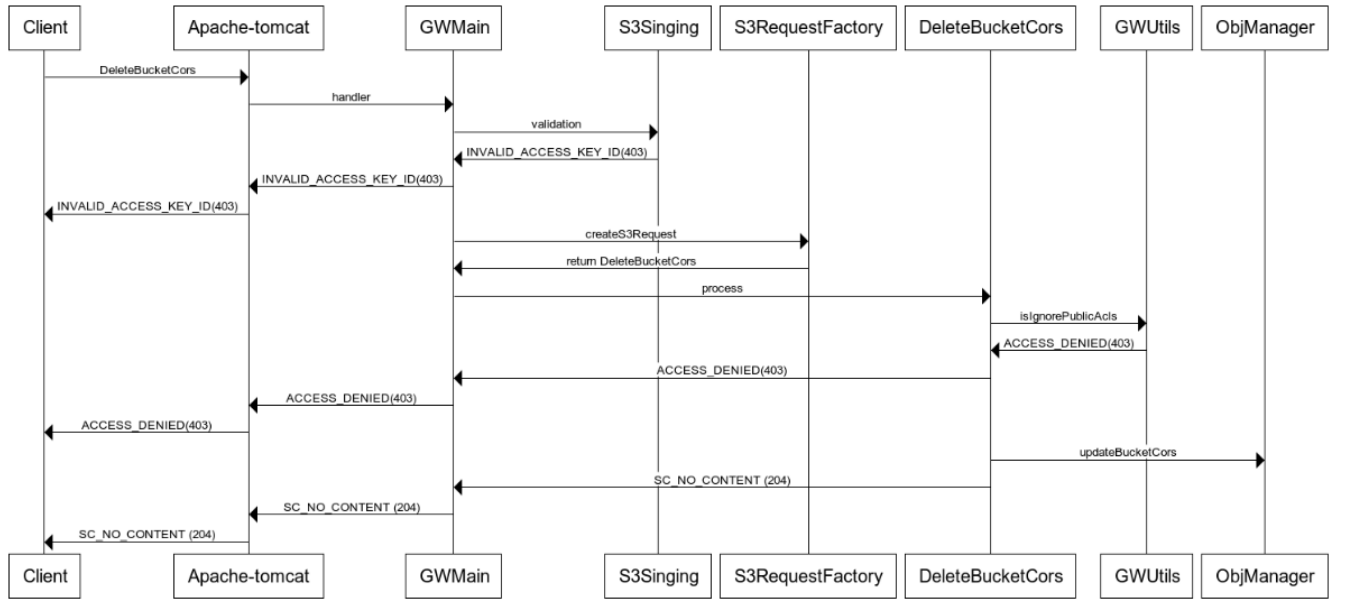
```
HTTP/1.1 204
```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공

Status Code	Description
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# DeleteBucketEncryption

## Overview

버킷에서 기본 암호화를 제거합니다.

## Request Syntax

```
DELETE /Bucket/?encryption HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /Bucket/?encryption HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 06 Sep 2017 12:00:00 GMT
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

## Sample Request

```
HTTP/1.1 204
```

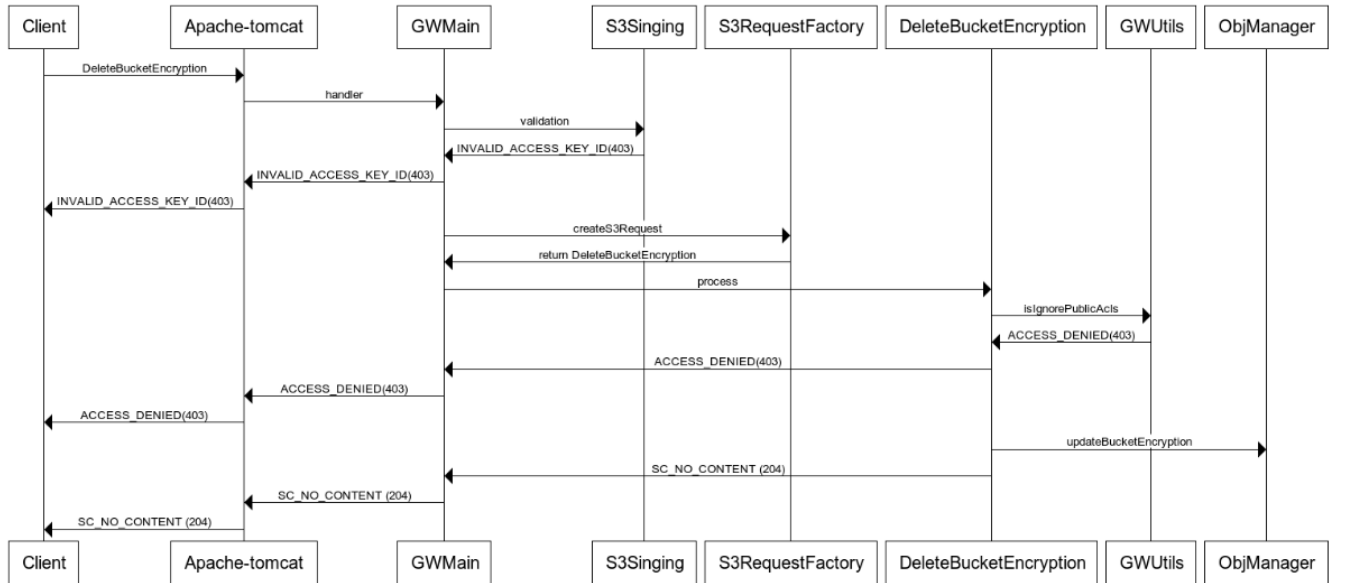
## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공



Status Code	Description
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# DeleteBucketLifecycle

## Overview

지정한 버킷에서 수명 주기 구성을 삭제합니다.

## Request Syntax

```
DELETE /Bucket/?lifecycle HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /ExampleBucket/?lifecycle HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 14 Dec 2011 05:37:16 GMT
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

## Sample Response

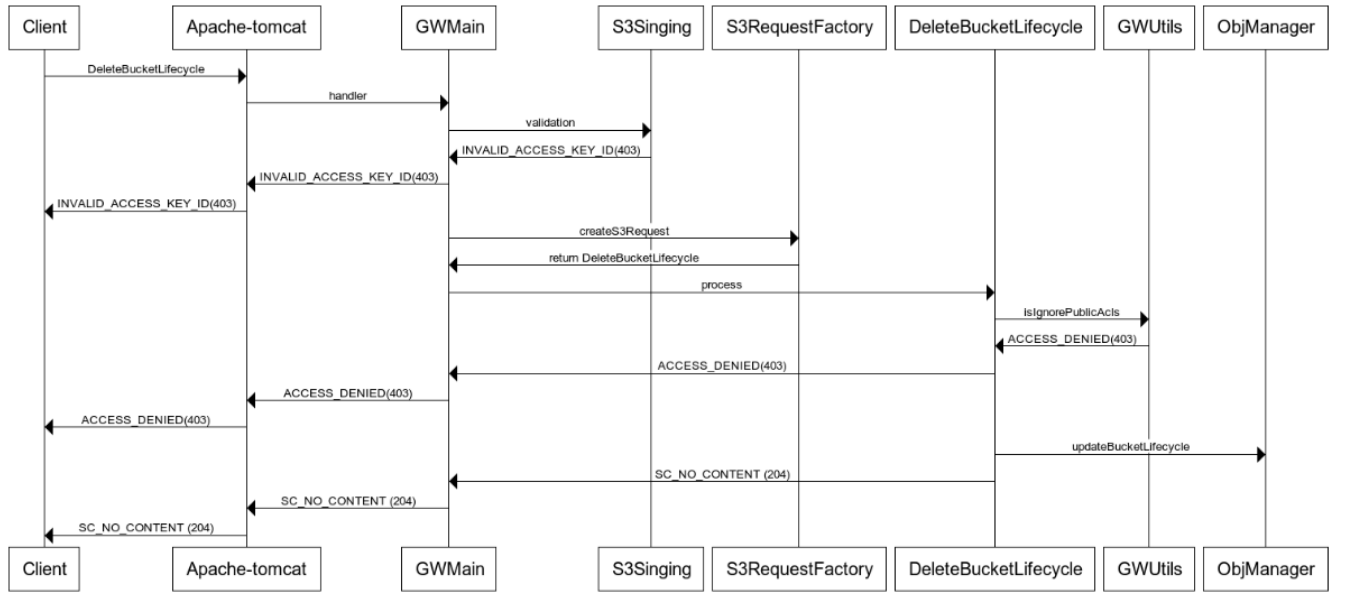
```
HTTP/1.1 204
```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공

Status Code	Description
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# DeleteBucketPolicy

## Overview

지정된 버킷의 정책을 삭제합니다.

## Request Syntax

```
DELETE /Bucket/?policy HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /ExampleBucket/?policy HTTP/1.1
Host: 192.168.111.111:8080
Date: Tue, 04 Apr 2010 20:34:56 GMT
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
```

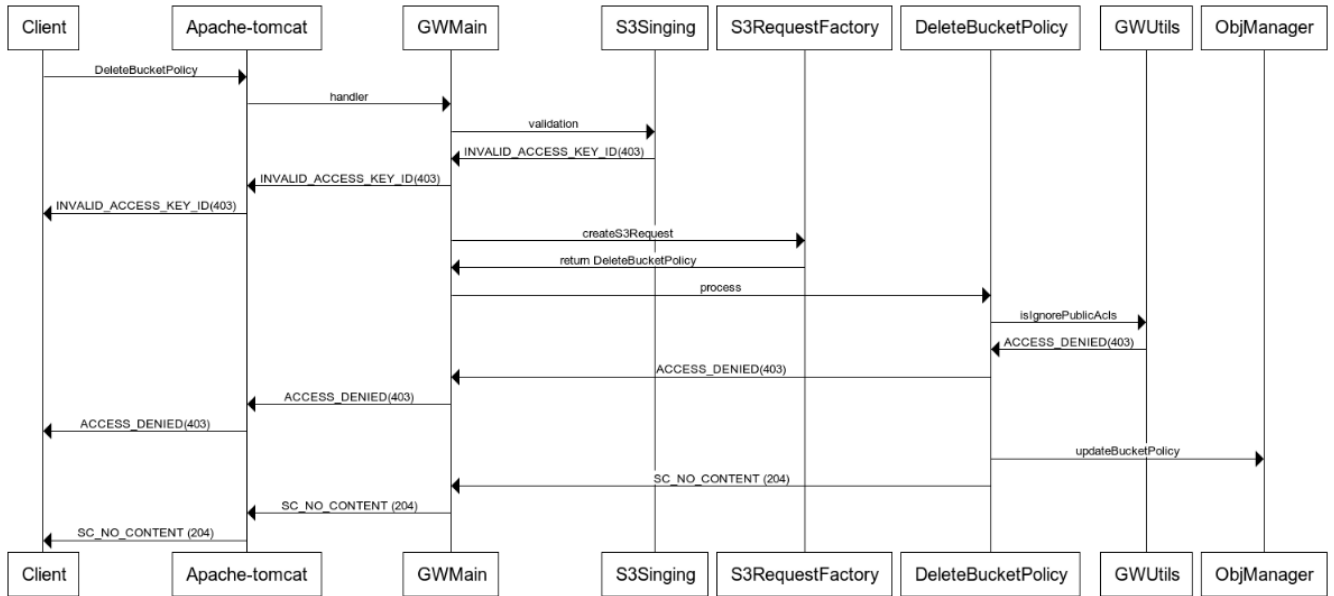
## Sample Request

```
HTTP/1.1 204
```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# DeleteBucketReplication

## Overview

버킷에서 복제 구성을 삭제합니다.

## Request Syntax

```
DELETE /Bucket/?replication HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /ExampleBucket/?replication HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 11 Feb 2015 05:37:16 GMT
20150211T171320Z
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

## Sample Response

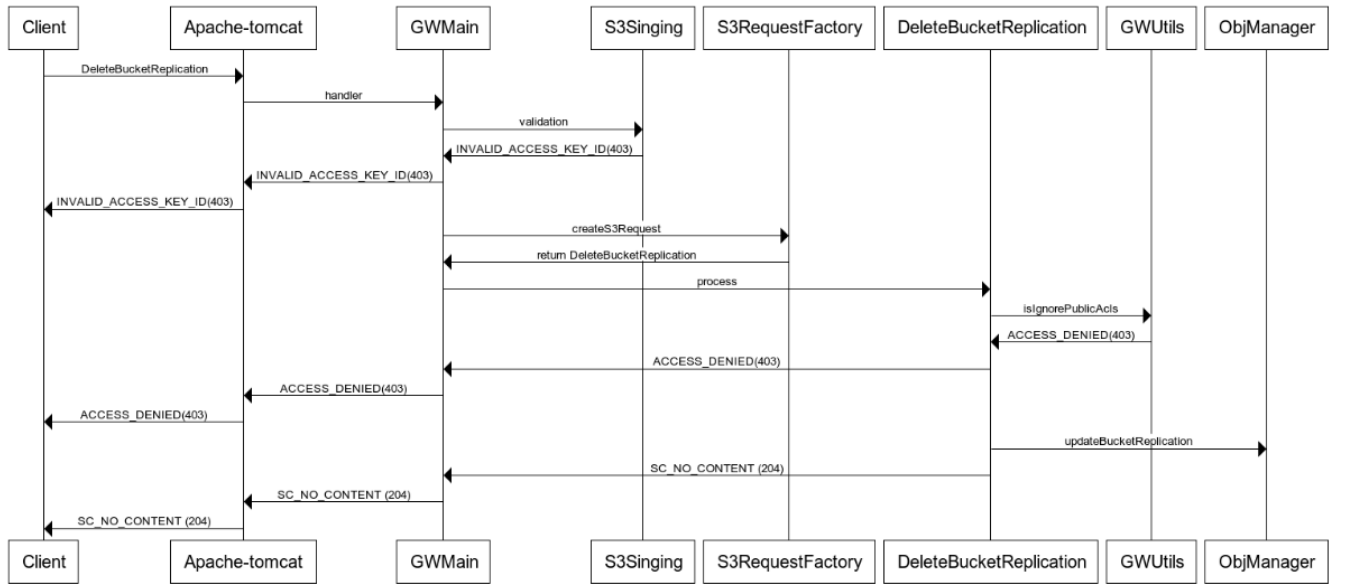
```
HTTP/1.1 204
```

## HTTP Status Code

Status Code	Description
-------------	-------------

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# DeleteBucketTagging

## Overview

버킷에서 태그를 삭제합니다.

## Request Syntax

```
DELETE /Bucket/?tagging HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /ExampleBucket/?tagging HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 14 Dec 2011 05:37:16 GMT
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

## Sample Response

```
HTTP/1.1 204
```

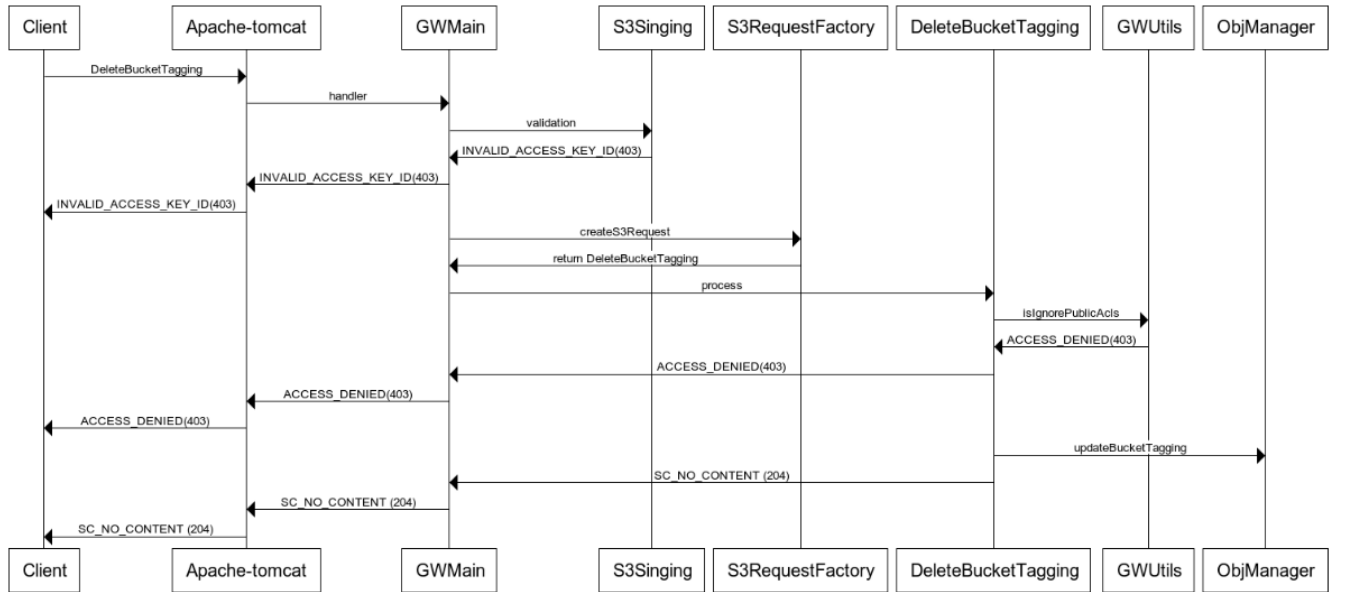
## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공



Status Code	Description
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# DeleteBucketWebsite

## Overview

버킷의 웹 사이트 구성을 제거합니다.

## Request Syntax

```
DELETE /Bucket/?website HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /ExampleBucket/?website HTTP/1.1
Host: 192.168.111.111:8080
Date: Thu, 27 Jan 2011 12:00:00 GMT
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
```

## Response Elements

## Sample Response

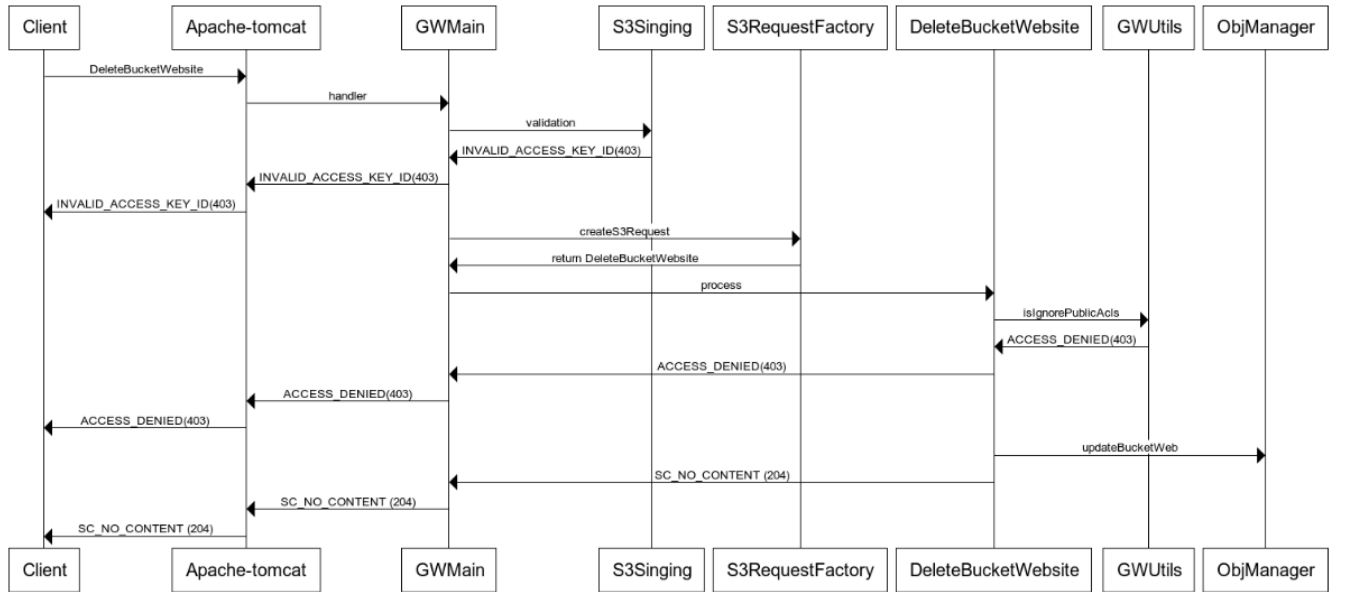
```
HTTP/1.1 204
```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공

Status Code	Description
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# DeleteObject

## Overview

개체의 null 버전(있는 경우)을 제거하고 삭제 마커를 삽입합니다. 이 마커는 개체의 최신 버전이 됩니다.

## Request Syntax

```
DELETE /Bucket/Key+?versionId=VersionId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-mfa: MFA
x-amz-request-payer: RequestPayer
x-amz-bypass-governance-retention: BypassGovernanceRetention
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /ExampleBucket/my-second-image.jpg HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
versionId	개체의 특정 버전을 참조하는 데 사용되는 VersionId입니다.	
x-amz-bypass-governance-retention	이 작업을 처리하기 위해 S3 Object Lock이 거버넌스 모드 제한을 무시해야 하는지 여부를 나타냅니다. 이 헤더를 사용하려면 s3:BypassGovernanceRetection 사용 권한이 있어야 합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-mfa	버전 관리가 MFA 삭제를 사용하도록 구성된 경우 버전 지정 개체를 영구적으로 삭제하는 데 필요합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
x-amz-delete-marker: DeleteMarker
x-amz-version-id: VersionId
x-amz-request-charged: RequestCharged
```

# Response Elements

Parameter Name	Description	Required
x-amz-delete-marker	영구적으로 삭제된 버전이 지정된 개체가 삭제 마커인지 (참) 또는 삭제 마커가 아닌지(거짓) 여부를 지정합니다.	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
x-amz-version-id	DELETE 작업의 결과로 생성된 마커의 버전 ID를 반환합니다.	

## Sample Response

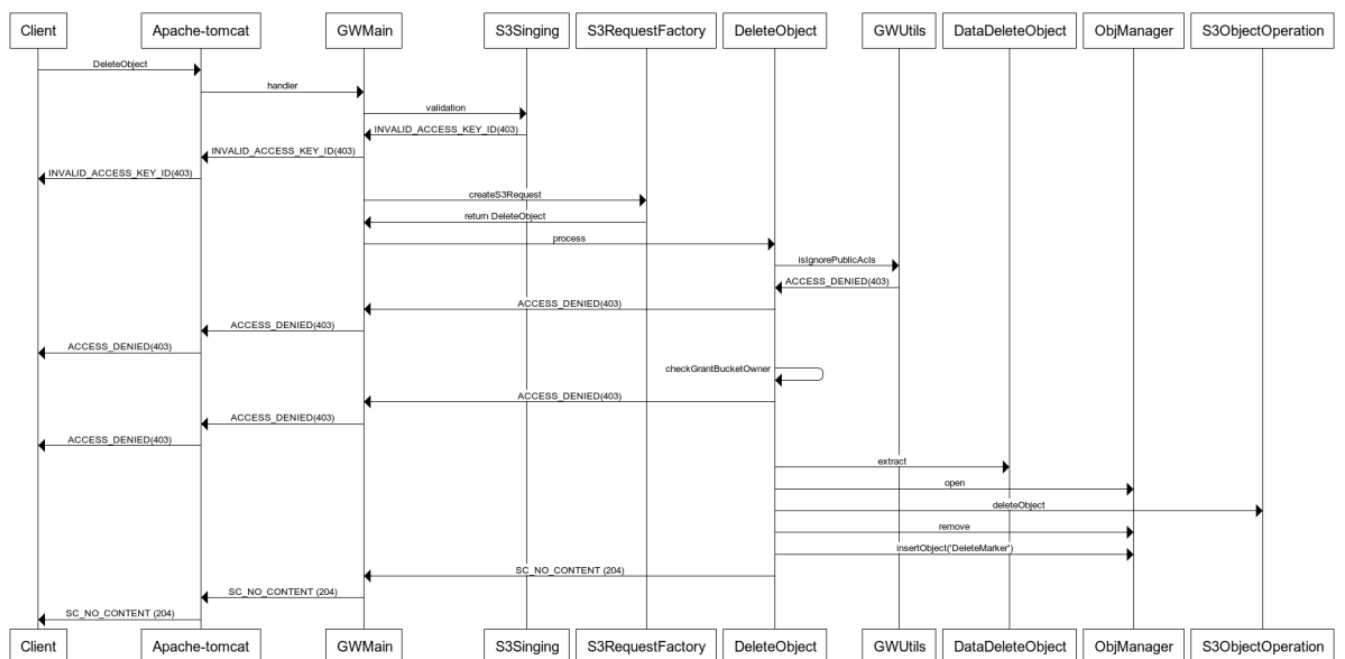
```

HTTP/1.1 204 NoContent
x-amz-id-2: LriYPLdm0dAiIfgSm/F1YsViT1LW94/xUQxMsF7xiEb1a0wiI0IxL+zbwZ163pt7
x-amz-request-id: 0A49CE4060975EAC
Date: Wed, 12 Oct 2009 17:50:00 GMT
Content-Length: 0
Connection: close
Server: KSAN-GW
    
```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# DeleteObjects

## Overview

이 작업을 통해 단일 HTTP 요청을 사용하여 버킷에서 여러 개체를 삭제할 수 있습니다.

## Request Syntax

```
POST /Bucket/?delete HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-mfa: MFA
x-amz-request-payer: RequestPayer
x-amz-bypass-governance-retention: BypassGovernanceRetention
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<Delete xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Object>
    <Key>string</Key>
    <VersionId>string</VersionId>
  </Object>
  ...
  <Quiet>boolean</Quiet>
</Delete>
```

## Sample Request

```
POST /ExampleBucket/?delete HTTP/1.1
Host: 192.168.111.111:8080
Accept: */*
x-amz-date: Wed, 30 Nov 2011 03:39:05 GMT
Content-MD5: p5/WA/oEr30qrEEL21PAqw==
Authorization: AWS AKIAIOSFODNN7EXAMPLE:W0qPYCLe6JwkZAD1ei6hp9XZiIee=
Content-Length: 125
Connection: Keep-Alive
<Delete>
  <Object>
    <Key>sample1.txt</Key>
  </Object>
  <Object>
    <Key>sample2.txt</Key>
  </Object>
</Delete>
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-bypass-governance-retention	이 작업을 처리하기 위해 S3 Object Lock이 거버넌스 모드 제한을 무시해야 하는지 여부를 나타냅니다. 이 헤더를 사용하려면 s3:BypassGovernanceRetection 사용 권한이 있어야 합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-mfa	버전 관리가 MFA 삭제를 사용하도록 구성된 경우 버전 지정 개체를 영구적으로 삭제하는 데 필요합니다.	

Parameter Name	Description	Required
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

# Request Body

```
<Delete xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Object>
    <Key>string</Key>
    <VersionId>string</VersionId>
    <Quiet>Boolean</Quiet> // Required : No
  </Object>
  ...
  <Quiet>boolean</Quiet>
</Delete>
```

# Response Syntax

```
HTTP/1.1 200
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<DeleteResult>
  <Deleted>
    <DeleteMarker>boolean</DeleteMarker>
    <DeleteMarkerVersionId>string</DeleteMarkerVersionId>
    <Key>string</Key>
    <VersionId>string</VersionId>
  </Deleted>
  ...
  <Error>
    <Code>string</Code>
    <Key>string</Key>
    <Message>string</Message>
    <VersionId>string</VersionId>
  </Error>
  ...
</DeleteResult>
```

# Response Elements

Parameter Name	Description	Required
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
DeleteResult	Response Syntax 참조	

# Sample Response

```
HTTP/1.1 200 OK
x-amz-id-2: 5h4FxSNCUS7wP5z92eGCWDshNpMnRuXvETa4HH3LvH6VAIr0jU7tH9kM7X+njXx
x-amz-request-id: A437B3B641629AEE
Date: Fri, 02 Dec 2011 01:53:42 GMT
Content-Type: application/xml
Server: AmazonS3
Content-Length: 251
<?xml version="1.0" encoding="UTF-8"?>
<DeleteResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
```

```

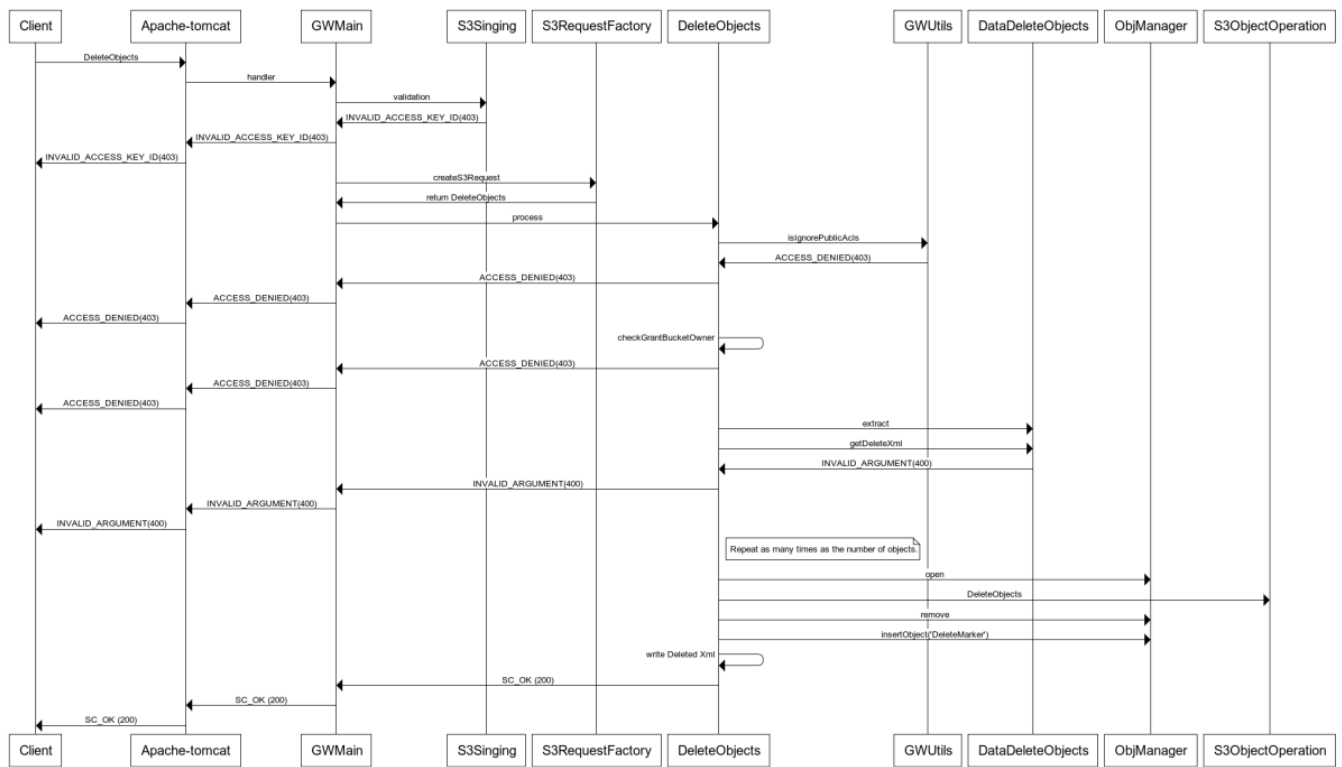
<Deleted>
  <Key>sample1.txt</Key>
</Deleted>
<Error>
  <Key>sample2.txt</Key>
  <Code>AccessDenied</Code>
  <Message>Access Denied</Message>
</Error>
</DeleteResult>

```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
400 (Invalid Argument)	Request Body Xml이 형식에 맞지 않음
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests





# DeleteObjectTagging

## Overview

지정한 개체에서 전체 태그 집합을 제거합니다.

## Request Syntax

```
DELETE /Bucket/Key?tagging&versionId=VersionId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
DELETE /ExampleBucket/exampleobject?tagging HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 25 Nov 2016 12:00:00 GMT
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
versionId	개체의 특정 버전을 참조하는 데 사용되는 VersionId입니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
x-amz-version-id: VersionId
```

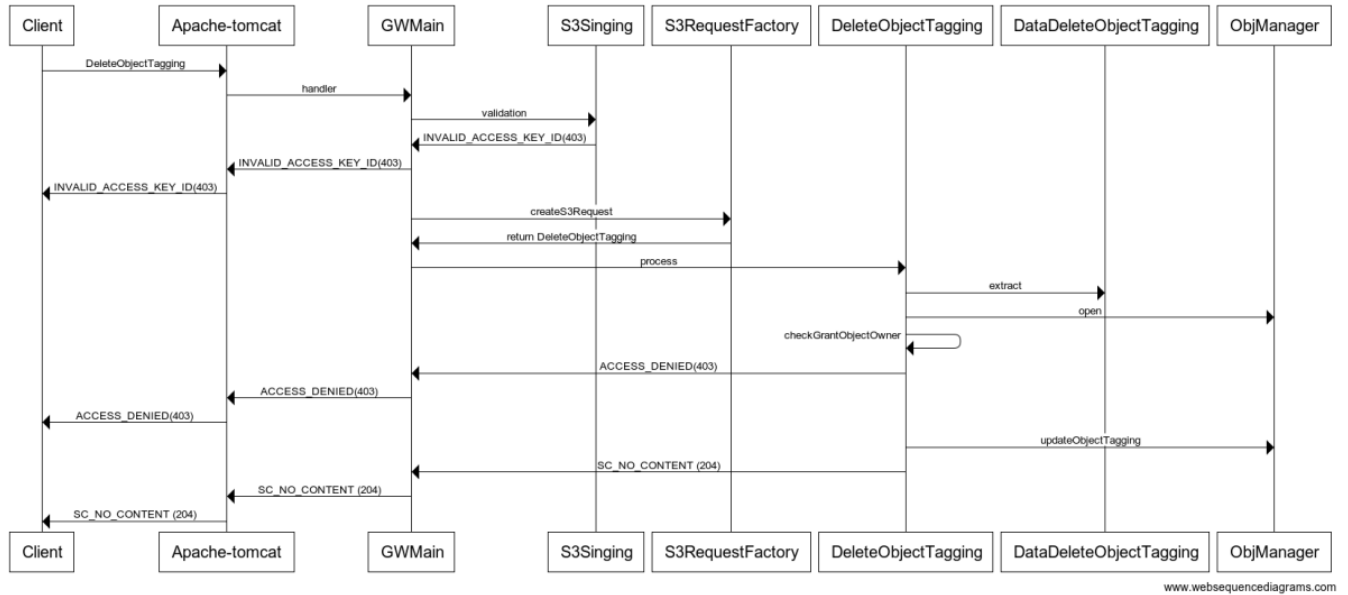
## Sample Request

```
HTTP/1.1 204 No Content
x-amz-versionid: VersionId
Date: Wed, 25 Nov 2016 12:00:00 GMT
Connection: close
Server: KSAN-GW
```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# DeletePublicAccessBlock

## Overview

S3 버킷에 대한 PublicAccessBlock 구성을 제거합니다.

## Request Syntax

```
DELETE /Bucket/?publicAccessBlock HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(엑세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 204
```

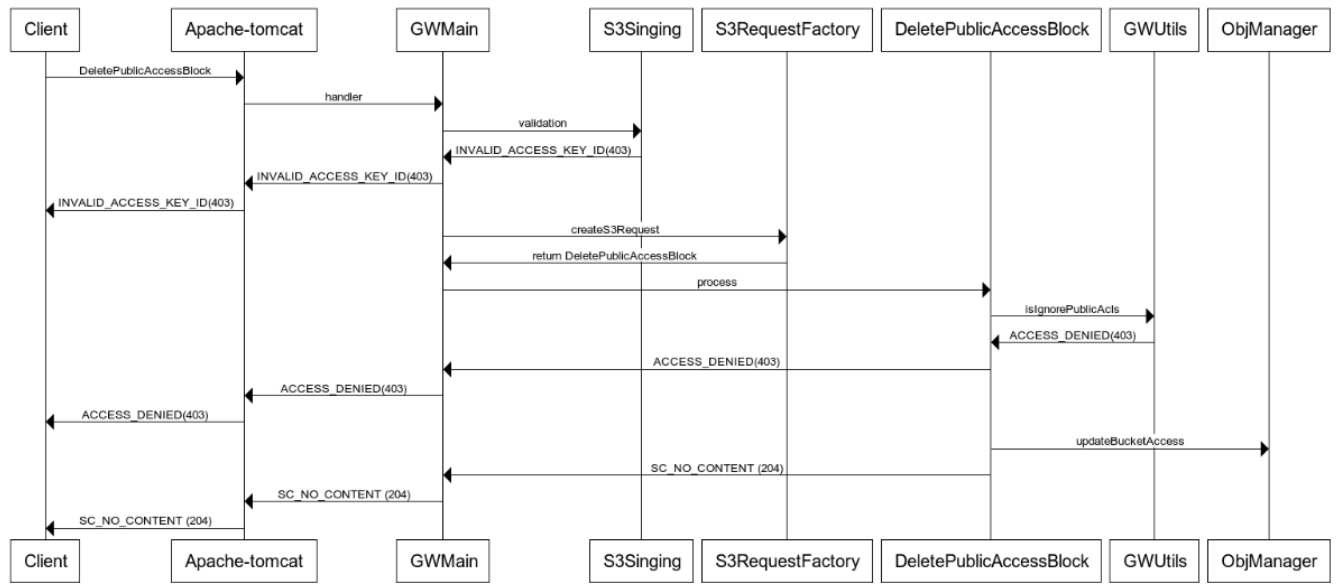
## Response Elements

## Sample Response

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# GetBucketAcl

## Overview

버킷의 ACL(액세스 제어 목록)을 반환합니다.

## Request Syntax

```
GET /Bucket/?acl HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?acl HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<AccessControlPolicy>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee>
        <DisplayName>string</DisplayName>
        <EmailAddress>string</EmailAddress>
        <ID>string</ID>
        <xsi:type>string</xsi:type>
        <URI>string</URI>
      </Grantee>
      <Permission>string</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>
```

# Response Elements

Parameter Name	Description	Required
AccessControlPolicy	REsponse Syntax 참조	Yes

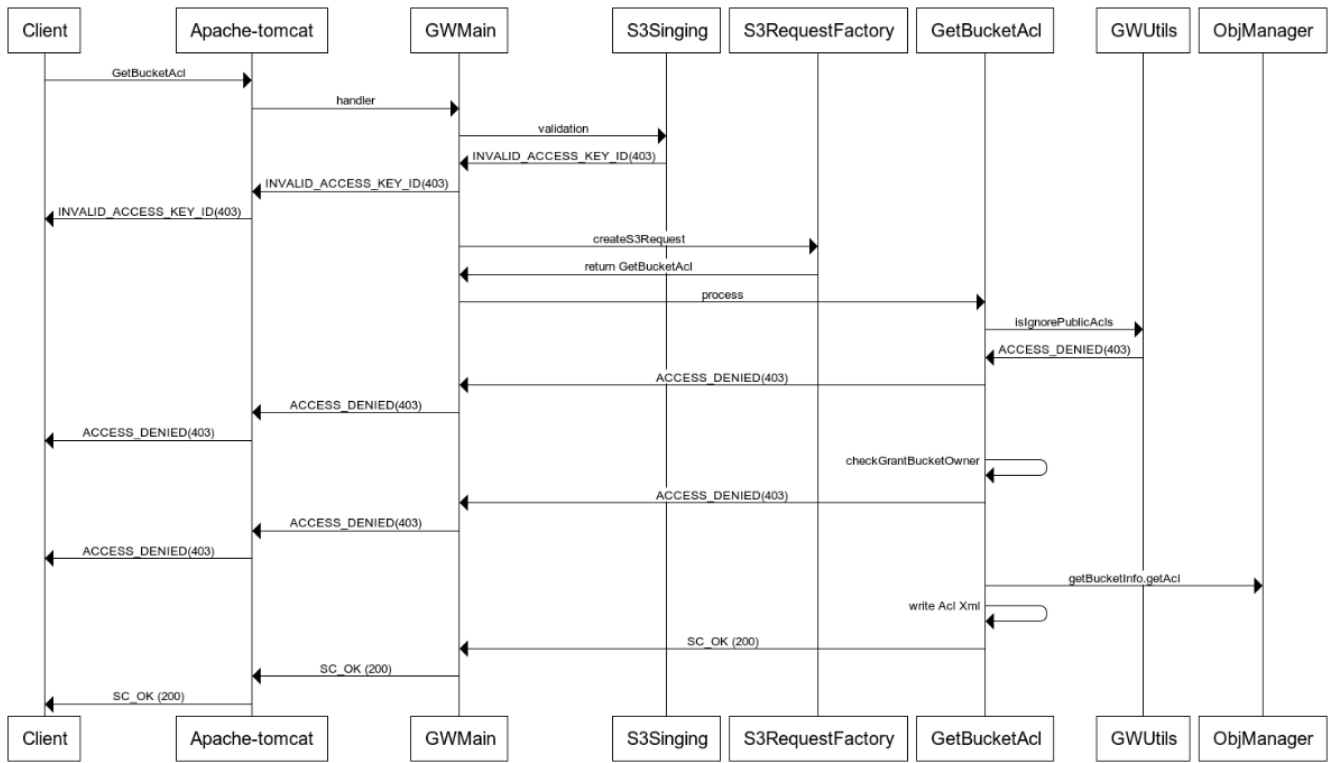
## Sample Response

```
HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TnqcoF8eFidJG9Z/2mkiDFu8yU9AS1ed40pIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
Content-Length: 124
Content-Type: text/plain
Connection: close
Server: KSAN-GW
<AccessControlPolicy>
  <Owner>
    <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>CustomersName@amazon.com</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="CanonicalUser">
        <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
        <DisplayName>CustomersName@amazon.com</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>
```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# GetBucketCors

## Overview

버킷에 대한 CORS(Cross-Origin Resource Sharing) 집합을 반환합니다.

## Request Syntax

```
GET /Bucket/?cors HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?cors HTTP/1.1
Host: 192.168.111.111:8080
Date: Tue, 13 Dec 2011 19:14:42 GMT
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration>
  <CORSRule>
    <AllowedHeader>string</AllowedHeader>
    ...
    <AllowedMethod>string</AllowedMethod>
    ...
    <AllowedOrigin>string</AllowedOrigin>
    ...
    <ExposeHeader>string</ExposeHeader>
    ...
    <ID>string</ID>
    <MaxAgeSeconds>integer</MaxAgeSeconds>
  </CORSRule>
  ...
</CORSConfiguration>
```



# Response Elements

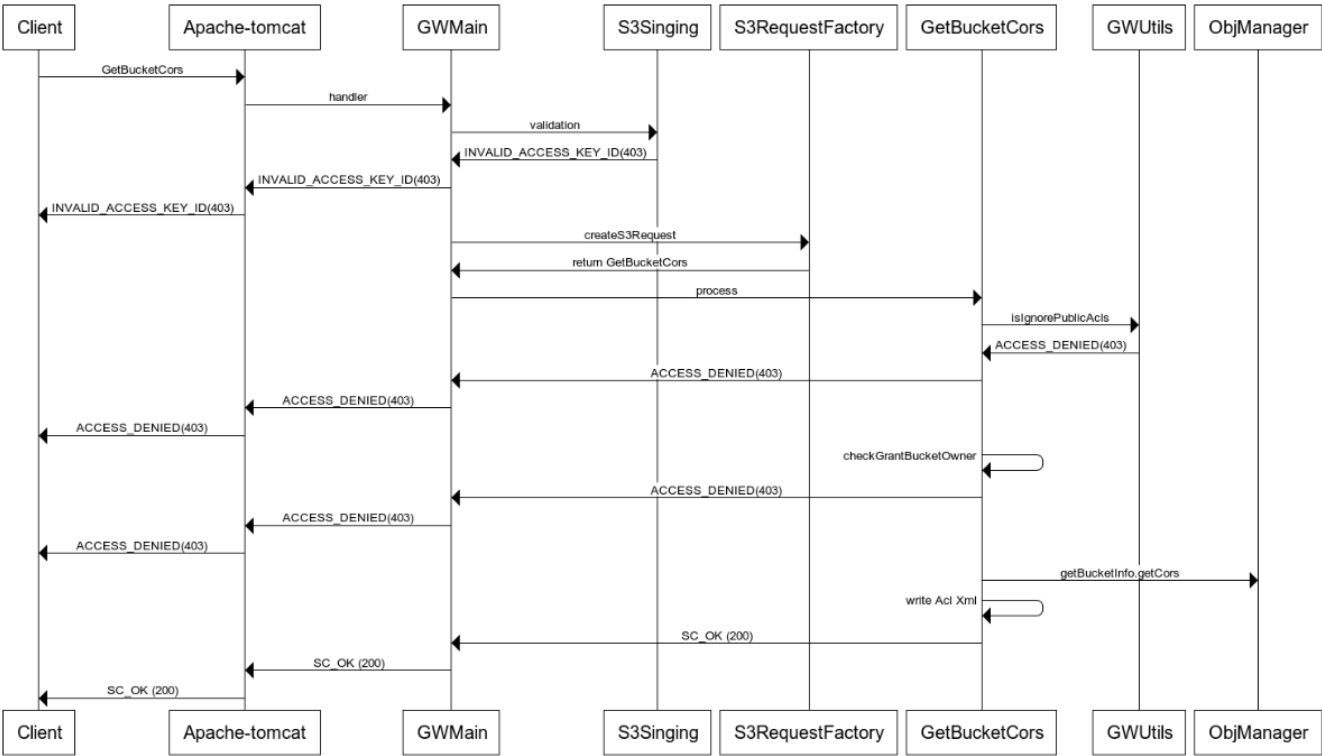
Parameter Name	Description	Required
CORSConfiguration	RResponse Syntax 참조	Yes

# Sample Response

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# GetBucketEncryption

## Overview

S3 버킷에 대한 기본 암호화 구성을 반환합니다.

## Request Syntax

```
GET /Bucket/?encryption HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?encryption HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 06 Sep 2017 12:00:00 GMT
Authorization: authorization string
Content-Length: length
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ServerSideEncryptionConfiguration>
  <Rule>
    <ApplyServerSideEncryptionByDefault>
      <KMSTMasterKeyID>string</KMSTMasterKeyID>
      <SSEAlgorithm>string</SSEAlgorithm>
    </ApplyServerSideEncryptionByDefault>
    <BucketKeyEnabled>boolean</BucketKeyEnabled>
  </Rule>
  ...
</ServerSideEncryptionConfiguration>
```

## Response Elements

Parameter Name	Description	Required
----------------	-------------	----------

Parameter Name	Description	Required
ServerSideEncryptionConfiguration	RResponse Syntax 참조	Yes

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: kDmqsuw5FDmgLmxQaUkd9A4NJ/PIiE0c1rAU/ue2Yp60toXs4I5k5fqlwZsA6fV+wJQCzRRwygQ=
x-amz-request-id: 5D8706FCB2673B7D
Date: Wed, 06 Sep 2017 12:00:00 GMT
Transfer-Encoding: chunked
Server: AmazonS3

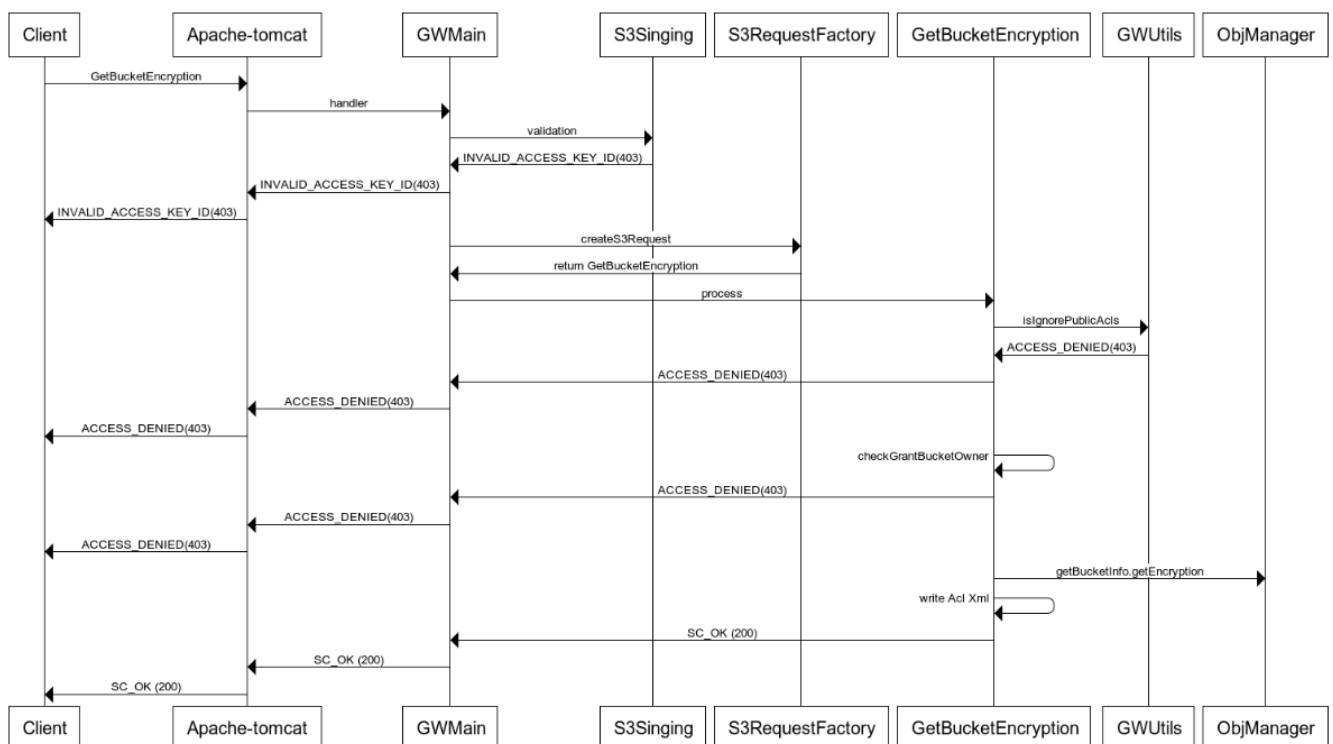
<ServerSideEncryptionConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ApplyServerSideEncryptionByDefault>
      <SSEAlgorithm>aws:kms</SSEAlgorithm>
      <KMSEMasterKeyID>arn:aws:kms:us-east-1:1234:5678example</KMSEMasterKeyID>
    </ApplyServerSideEncryptionByDefault>
  </Rule>
</ServerSideEncryptionConfiguration>

```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# GetBucketLifecycleConfiguration

## Overview

버킷에 설정된 수명 주기 구성 정보를 반환합니다.

## Request Syntax

```
GET /Bucket/?lifecycle HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?lifecycle HTTP/1.1
Host: 192.168.111.111:8080
x-amz-date: Thu, 15 Nov 2012 00:17:21 GMT
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration>
  <Rule>
    <AbortIncompleteMultipartUpload>
      <DaysAfterInitiation>integer</DaysAfterInitiation>
    </AbortIncompleteMultipartUpload>
    <Expiration>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <ExpiredObjectDeleteMarker>boolean</ExpiredObjectDeleteMarker>
    </Expiration>
    <Filter>
      <And>
        <ObjectSizeGreaterThan>long</ObjectSizeGreaterThan>
        <ObjectSizeLessThan>long</ObjectSizeLessThan>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
      </And>
    </Filter>
  </Rule>
</LifecycleConfiguration>
```

```

        </Tag>
        ...
    </And>
    <ObjectSizeGreaterThan>long</ObjectSizeGreaterThan>
    <ObjectSizeLessThan>long</ObjectSizeLessThan>
    <Prefix>string</Prefix>
    <Tag>
        <Key>string</Key>
        <Value>string</Value>
    </Tag>
</Filter>
<ID>string</ID>
<NoncurrentVersionExpiration>
    <NewerNoncurrentVersions>integer</NewerNoncurrentVersions>
    <NoncurrentDays>integer</NoncurrentDays>
</NoncurrentVersionExpiration>
<NoncurrentVersionTransition>
    <NewerNoncurrentVersions>integer</NewerNoncurrentVersions>
    <NoncurrentDays>integer</NoncurrentDays>
    <StorageClass>string</StorageClass>
</NoncurrentVersionTransition>
...
<Prefix>string</Prefix>
<Status>string</Status>
<Transition>
    <Date>timestamp</Date>
    <Days>integer</Days>
    <StorageClass>string</StorageClass>
</Transition>
...
</Rule>
...
</LifecycleConfiguration>

```

## Response Elements

Parameter Name	Description	Required
LifecycleConfiguration	RResponse Syntax 참조	Yes

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4RyTmXa3rPi4hklTXouTf0hccUjo0iCPjz6FnfIutBj3M7fPGLW02SEWp
x-amz-request-id: 51991C342C575321
Date: Thu, 15 Nov 2012 00:17:23 GMT
Server: KSAN-GW
Content-Length: 358

<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ID>Archive and then delete rule</ID>
    <Prefix>projectdocs</Prefix>
    <Status>Enabled</Status>
    <Transition>
      <Days>30</Days>
      <StorageClass>STANDARD_IA</StorageClass>
    </Transition>
    <Transition>
      <Days>365</Days>
      <StorageClass>GLACIER</StorageClass>
    </Transition>
  </Rule>
  <Expiration>

```

```

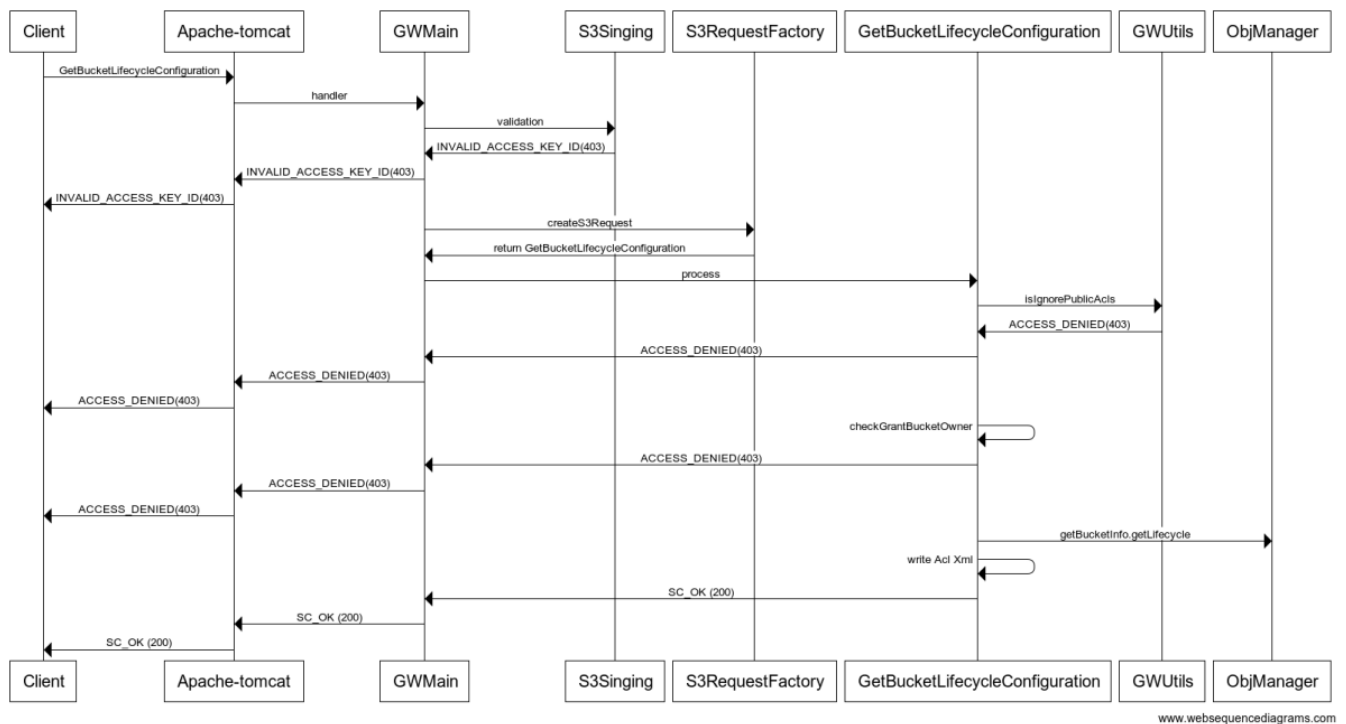
<Days>3650</Days>
</Expiration>
</Rule>
</LifecycleConfiguration>

```

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# GetBucketLocation

## Overview

버킷이 있는 영역을 반환합니다.

## Request Syntax

```
GET /Bucket/?location HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?location HTTP/1.1
Host: 192.168.111.111:8080
Date: Tue, 09 Oct 2007 20:26:04 +0000
Authorization: signatureValue
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<LocationConstraint>
  <LocationConstraint>string</LocationConstraint>
</LocationConstraint>
```

## Response Elements

Parameter Name	Description	Required
LocationConstraint	RResponse Syntax 참조	Yes

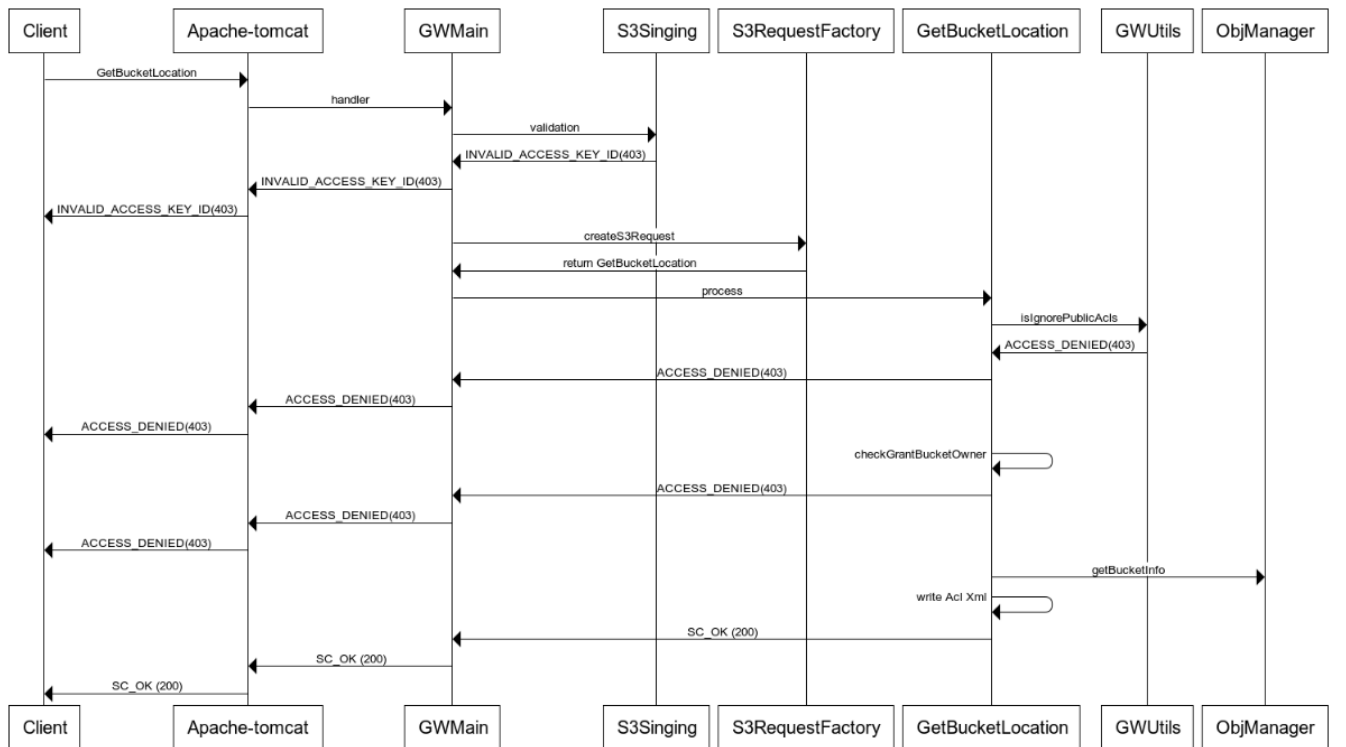
## Sample Response

```
<?xml version="1.0" encoding="UTF-8"?>
<LocationConstraint xmlns="http://s3.amazonaws.com/doc/2006-03-01/">Europe</LocationConstraint>
```

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests





# GetBucketPolicy

## Overview

지정한 버킷의 정책을 반환합니다.

## Request Syntax

```
GET /Bucket/?policy HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?policy HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
{ Policy in JSON format }
```

## Response Elements

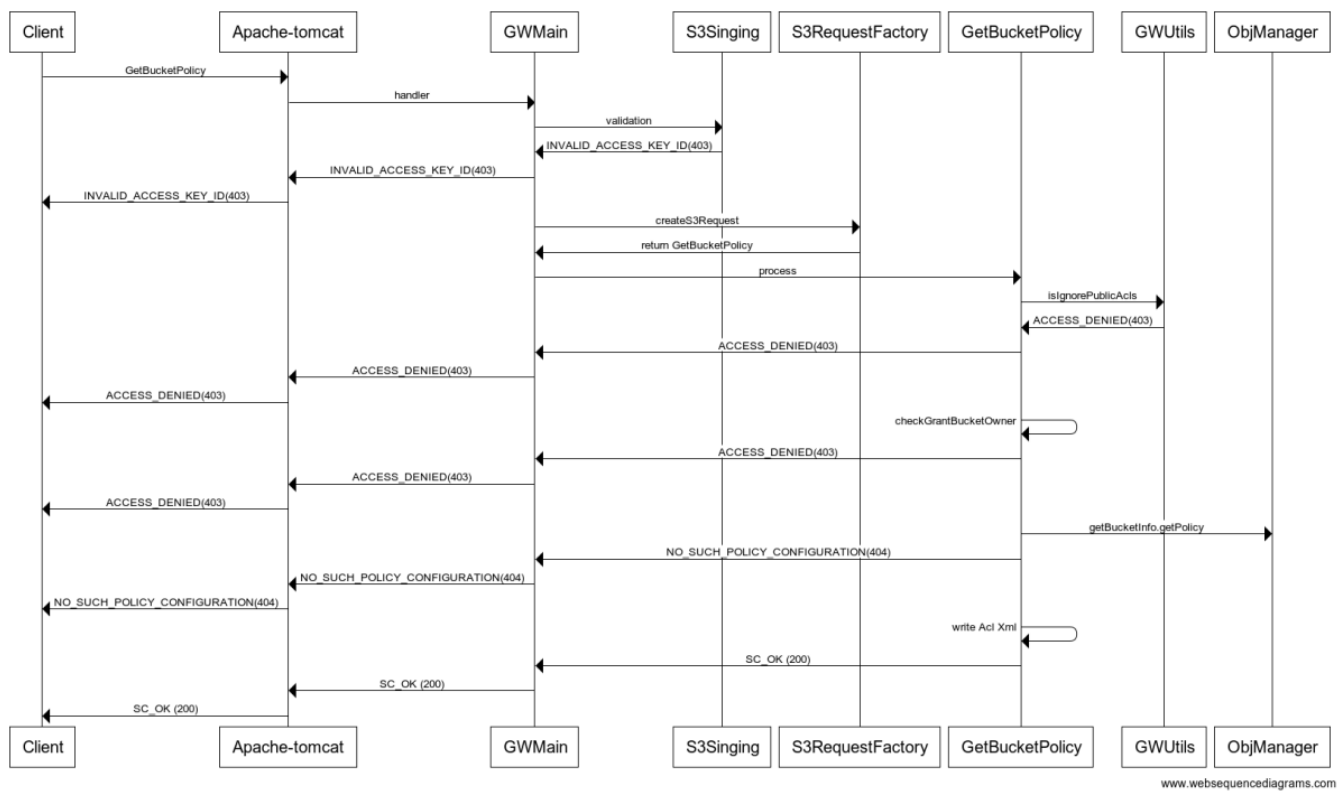
Parameter Name	Description	Required
Policy	RResponse Syntax 참조	Yes

## Sample Response

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# GetBucketPolicyStatus

## Overview

S3 버킷의 정책 상태를 검색하여 버킷이 공용인지 여부를 나타냅니다.

## Request Syntax

```
GET /Bucket/?policyStatus HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?policyStatus HTTP/1.1
Host: 192.168.111.111:8080
x-amz-date: <Thu, 15 Nov 2016 00:17:21 GMT>
Authorization: <signatureValue>
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<PolicyStatus>
  <IsPublic>boolean</IsPublic>
</PolicyStatus>
```

## Response Elements

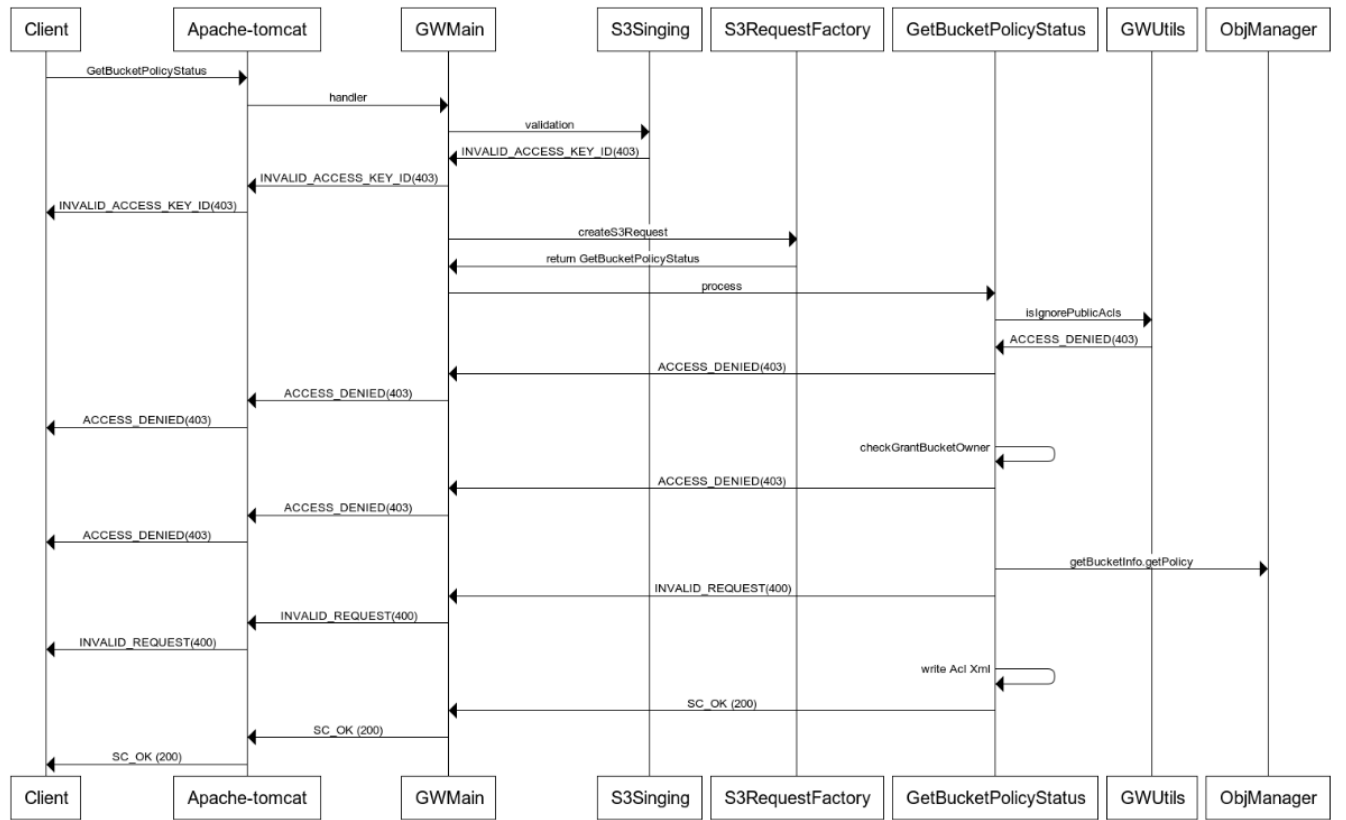
Parameter Name	Description	Required
PolicyStatus	RResponse Syntax 참조	Yes

## Sample Response

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# GetBucketReplication

## Overview

버킷의 복제 구성을 반환합니다.

## Request Syntax

```
GET /Bucket/?replication HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?replication HTTP/1.1
Host: 192.168.111.111:8080
Date: Tue, 10 Feb 2015 00:17:21 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ReplicationConfiguration>
  <Role>string</Role>
  <Rule>
    <DeleteMarkerReplication>
      <Status>string</Status>
    </DeleteMarkerReplication>
    <Destination>
      <AccessControlTranslation>
        <Owner>string</Owner>
      </AccessControlTranslation>
      <Account>string</Account>
      <Bucket>string</Bucket>
      <EncryptionConfiguration>
        <ReplicaKmsKeyID>string</ReplicaKmsKeyID>
      </EncryptionConfiguration>
      <Metrics>
        <EventThreshold>
          <Minutes>integer</Minutes>
        </EventThreshold>
      </Metrics>
    </Destination>
  </Rule>
</ReplicationConfiguration>
```

```

        </EventThreshold>
        <Status>string</Status>
    </Metrics>
    <ReplicationTime>
        <Status>string</Status>
        <Time>
            <Minutes>integer</Minutes>
        </Time>
    </ReplicationTime>
    <StorageClass>string</StorageClass>
</Destination>
<ExistingObjectReplication>
    <Status>string</Status>
</ExistingObjectReplication>
<Filter>
    <And>
        <Prefix>string</Prefix>
        <Tag>
            <Key>string</Key>
            <Value>string</Value>
        </Tag>
        ...
    </And>
    <Prefix>string</Prefix>
    <Tag>
        <Key>string</Key>
        <Value>string</Value>
    </Tag>
</Filter>
<ID>string</ID>
<Prefix>string</Prefix>
<Priority>integer</Priority>
<SourceSelectionCriteria>
    <ReplicaModifications>
        <Status>string</Status>
    </ReplicaModifications>
    <SseKmsEncryptedObjects>
        <Status>string</Status>
    </SseKmsEncryptedObjects>
</SourceSelectionCriteria>
    <Status>string</Status>
</Rule>
...
</ReplicationConfiguration>

```

## Response Elements

Parameter Name	Description	Required
ReplicationConfiguration	RResponse Syntax 참조	Yes

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4RyTmXa3rPi4hklTXouTf0hccUjo0iCPjz6FnfIutBj3M7fPGLW02SEWp
x-amz-request-id: 51991C342example
Date: Tue, 10 Feb 2015 00:17:23 GMT
Server: AmazonS3
Content-Length: contentlength

<?xml version="1.0" encoding="UTF-8"?>
<ReplicationConfiguration>
    <Role>arn:aws:iam::35667example:role/CrossRegionReplicationRoleForS3</Role>
</Role>

```

```

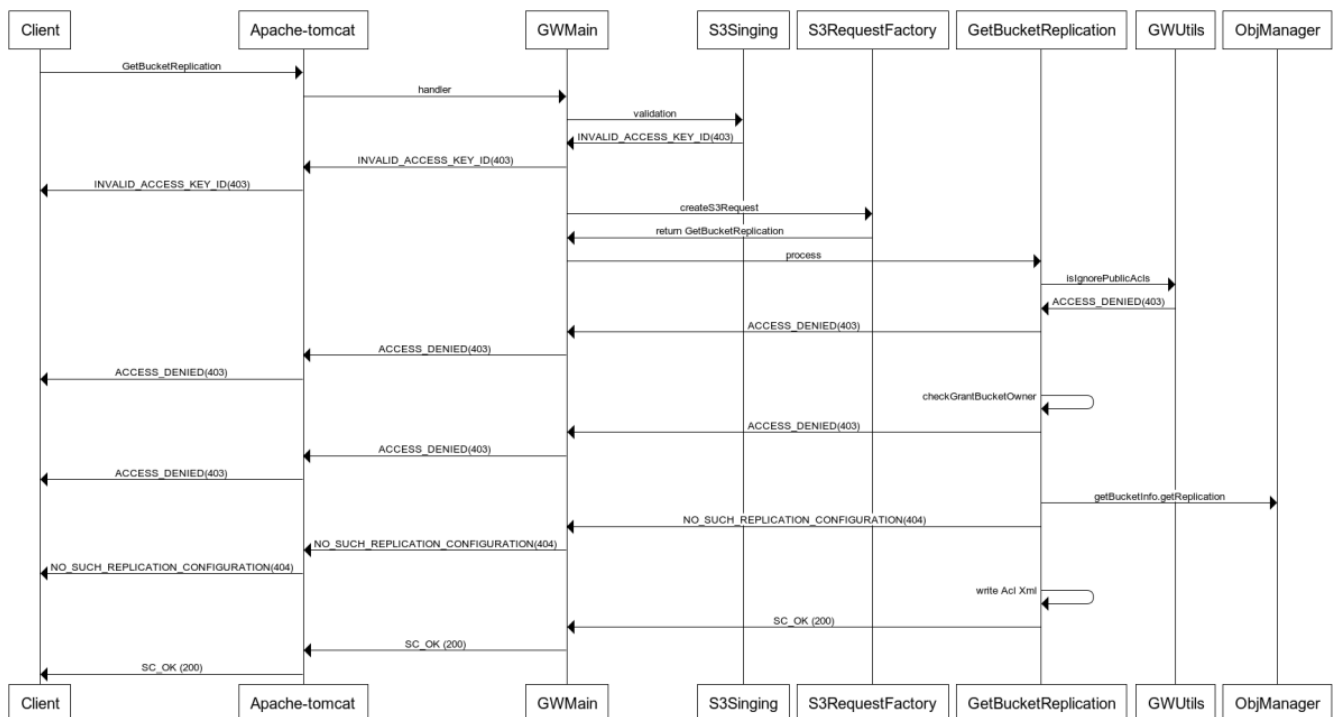
<ID>rule1</ID>
<Status>Enabled</Status>
<Priority>1</Priority>
<DeleteMarkerReplication>
  <Status>Disabled</Status>
</DeleteMarkerReplication>
<Filter>
  <And>
    <Prefix>TaxDocs</Prefix>
    <Tag>
      <Key>key1</Key>
      <Value>value1</Value>
    </Tag>
    <Tag>
      <Key>key1</Key>
      <Value>value1</Value>
    </Tag>
  </And>
</Filter>
<Destination>
  <Bucket>arn:aws:s3:::exampletargetbucket</Bucket>
</Destination>
</Rule>
</ReplicationConfiguration>

```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# GetBucketTagging

## Overview

버킷과 연결된 태그 집합을 반환합니다.

## Request Syntax

```
GET /Bucket/?tagging HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?tagging HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<Tagging>
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>
```

## Response Elements

Parameter Name	Description	Required
Tagging	RResponse Syntax 참조	Yes

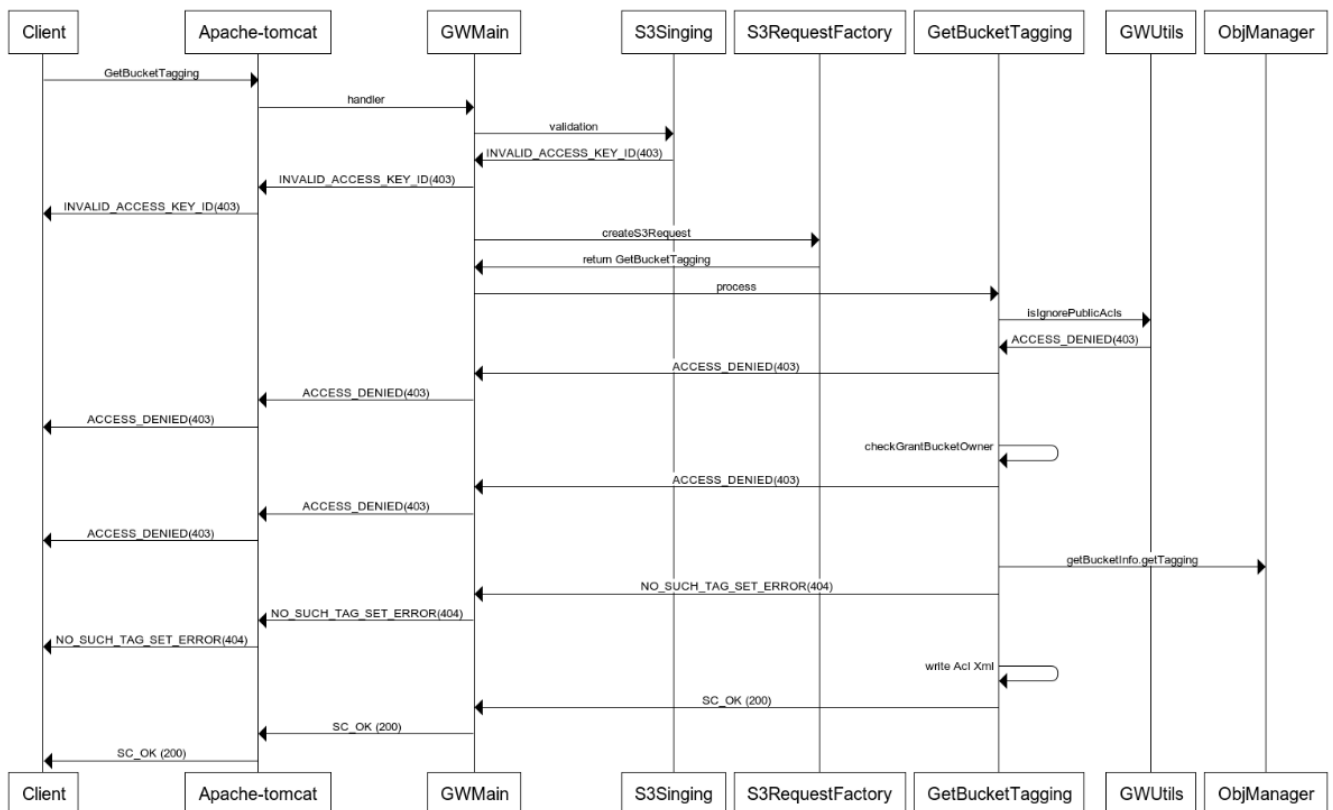


# Sample Response

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# GetBucketVersioning

## Overview

버킷의 버전 상태를 반환합니다.

## Request Syntax

```
GET /Bucket/?versioning HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?versioning HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<VersioningConfiguration>
  <Status>string</Status>
  <MfaDelete>string</MfaDelete>
</VersioningConfiguration>
```

## Response Elements

Parameter Name	Description	Required
VersioningConfiguration	REsponse Syntax 참조	Yes

## Sample Response

- Enabled

```
<VersioningConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Status>Enabled</Status>
</VersioningConfiguration>
```

- Suspended

```
<VersioningConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Status>Suspended</Status>
</VersioningConfiguration>
```

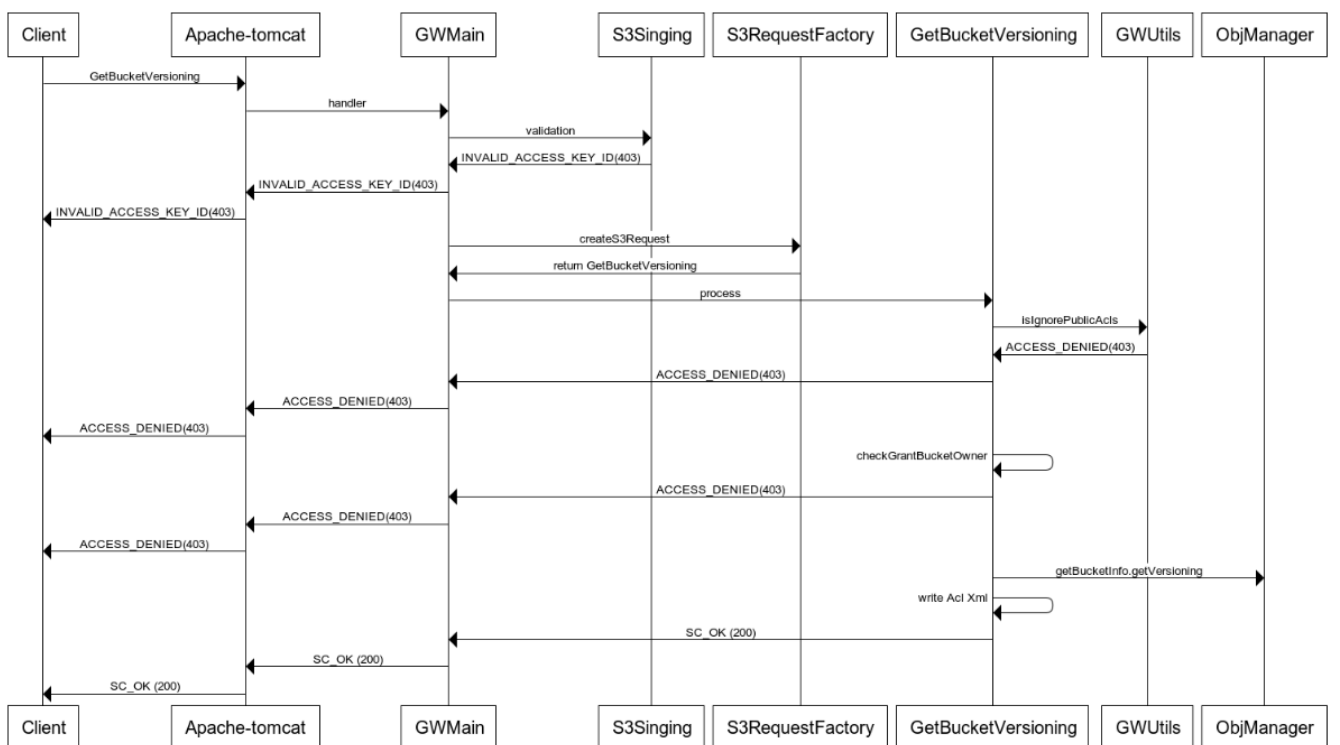
- Never Enabled(or Suspended) versioning on a bucket

```
<VersioningConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# GetBucketWebsite

## Overview

버킷의 웹 사이트 구성을 반환합니다.

## Request Syntax

```
GET /Bucket/?website HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?website HTTP/1.1
Host: example-bucket.s3.<Region>.amazonaws.com
Date: Thu, 27 Jan 2011 00:49:20 GMT
Authorization: AWS AKIAIOSFODNN7EXAMPLE:n0Nhek72Ufg/u7Sm5C1dqRLs8XX=
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<WebsiteConfiguration>
  <RedirectAllRequestsTo>
    <HostName>string</HostName>
    <Protocol>string</Protocol>
  </RedirectAllRequestsTo>
  <IndexDocument>
    <Suffix>string</Suffix>
  </IndexDocument>
  <ErrorDocument>
    <Key>string</Key>
  </ErrorDocument>
  <RoutingRules>
    <RoutingRule>
      <Condition>
        <HttpErrorCodeReturnedEquals>string</HttpErrorCodeReturnedEquals>
        <KeyPrefixEquals>string</KeyPrefixEquals>
      </Condition>
      <Redirect>
```

```

        <HostName>string</HostName>
        <HttpRedirectCode>string</HttpRedirectCode>
        <Protocol>string</Protocol>
        <ReplaceKeyPrefixWith>string</ReplaceKeyPrefixWith>
        <ReplaceKeyWith>string</ReplaceKeyWith>
    </Redirect>
</RoutingRule>
</RoutingRules>
</WebsiteConfiguration>

```

## Response Elements

Parameter Name	Description	Required
WebsiteConfiguration	RResponse Syntax 참조	Yes

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMgUAdQkf3ShJT00pXUueF6QKo
x-amz-request-id: 3848CD259D811111
Date: Thu, 27 Jan 2011 00:49:26 GMT
Content-Length: 240
Content-Type: application/xml
Transfer-Encoding: chunked
Server: KSan-GW

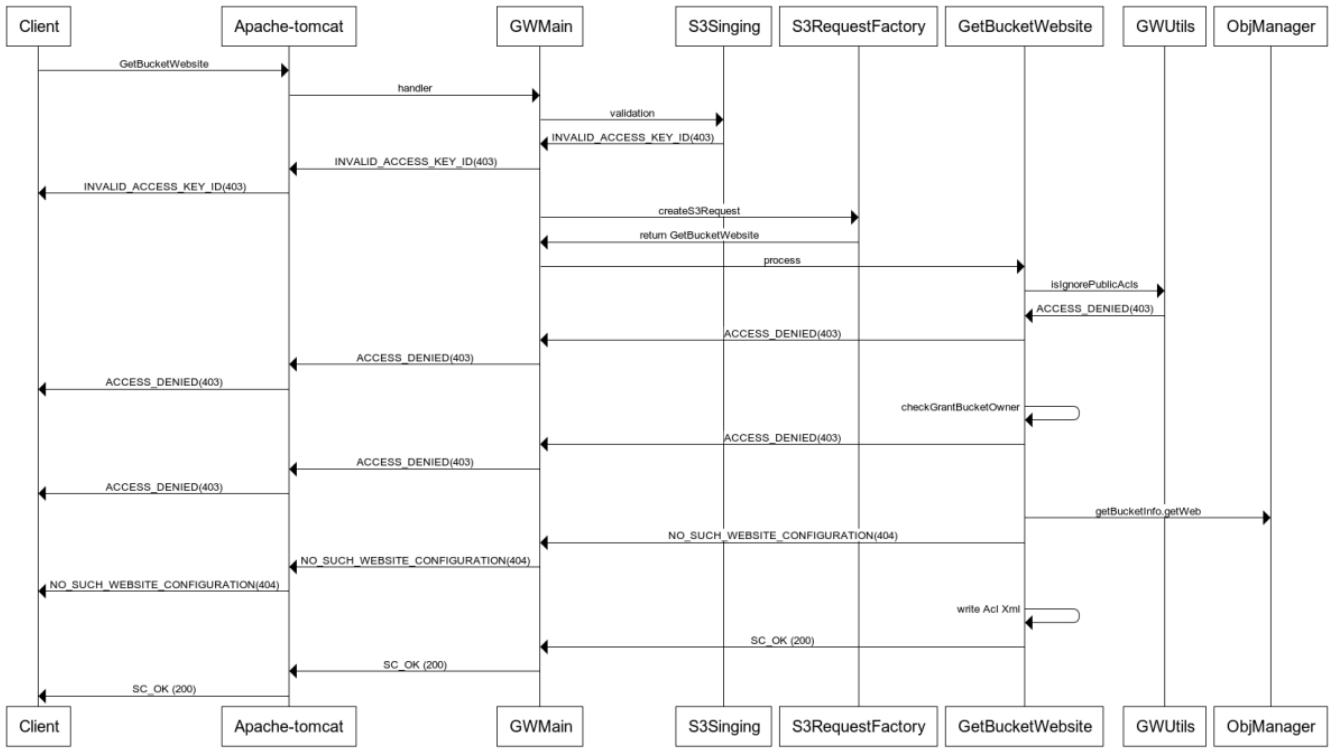
<?xml version="1.0" encoding="UTF-8"?>
<WebsiteConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <IndexDocument>
    <Suffix>index.html</Suffix>
  </IndexDocument>
  <ErrorDocument>
    <Key>404.html</Key>
  </ErrorDocument>
</WebsiteConfiguration>

```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# GetObject

## Overview

S3에서 개체를 가져옵니다. GET를 사용하려면 개체에 대한 READ 권한이 있어야 합니다.

## Request Syntax

```
GET /Bucket/Key?response-cache-control=ResponseCacheControl&response-content-
disposition=ResponseContentDisposition&response-content-encoding=ResponseContentEncoding&response-content-
language=ResponseContentLanguage&response-content-type=ResponseContentType&response-
expires=ResponseExpires&versionId=VersionId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
If-Match: IfMatch
If-Modified-Since: IfModifiedSince
If-None-Match: IfNoneMatch
If-Unmodified-Since: IfUnmodifiedSince
Range: Range
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/my-image.jpg HTTP/1.1
Host: bucket.s3.<Region>.amazonaws.com
Date: Mon, 3 Oct 2016 22:32:00 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
versionId	개체의 특정 버전을 참조하는 데 사용되는 VersionId입니다.	
If-Match	해당 엔티티 태그(ETAG)가 지정된 개체와 동일한 경우에만 개체를 반환하고 그렇지 않으면 412(전제 조건 실패)를 반환합니다.	
If-Modified-Since	지정된 시간 이후 개체를 수정한 경우에만 개체를 반환하고, 그렇지 않으면 304(수정되지 않음)를 반환합니다.	
If-None-Match	해당 엔티티 태그(ETAG)가 지정된 개체와 다른 경우에만 개체를 반환하고, 그렇지 않으면 수정되지 않은 304를 반환합니다.	
If-Unmodified-Since	지정된 시간 이후 개체를 수정하지 않은 경우에만 개체를 반환하고, 그렇지 않으면 412(전제 조건 실패)를 반환합니다.	
Range	개체의 지정된 범위 바이트를 다운로드합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

Parameter Name	Description	Required
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	
x-amz-server-side-encryption-customer-algorithm	개체를 암호화할 때 사용할 알고리즘(예: AES256)을 지정합니다.	
x-amz-server-side-encryption-customer-key	데이터 암호화에 사용할 S3에 대해 고객이 제공한 암호화 키를 지정합니다.	
x-amz-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 128비트 MD5 다이제스트를 지정합니다.	

## Request Body

## Response Syntax

```

HTTP/1.1 200
x-amz-delete-marker: DeleteMarker
accept-ranges: AcceptRanges
x-amz-expiration: Expiration
x-amz-restore: Restore
Last-Modified: LastModified
Content-Length: ContentLength
ETag: ETag
x-amz-missing-meta: MissingMeta
x-amz-version-id: VersionId
Cache-Control: CacheControl
Content-Disposition: ContentDisposition
Content-Encoding: ContentEncoding
Content-Language: ContentLanguage
Content-Range: ContentRange
Content-Type: ContentType
Expires: Expires
x-amz-website-redirect-location: WebsiteRedirectLocation
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-storage-class: StorageClass
x-amz-request-charged: RequestCharged
x-amz-replication-status: ReplicationStatus
x-amz-mp-parts-count: PartsCount
x-amz-tagging-count: TagCount
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
x-amz-object-lock-legal-hold: ObjectLockLegalHoldStatus


```

Body

## Response Elements

Parameter Name	Description	Required
accept-ranges	바이트 범위가 지정되었음을 나타냅니다.	
Cache-Control	요청/응답 체인을 따라 캐싱 동작을 지정합니다.	
Content-Disposition	개체에 대한 프레젠테이션 정보를 지정합니다.	
Content-Encoding	개체에 적용된 콘텐츠 인코딩과 콘텐츠 유형 헤더 필드가 참조하는 미디어 유형을 가져오기 위해 적용해야 하는 디코딩 메커니즘을 지정합니다.	



Parameter Name	Description	Required
Content-Language	콘텐츠가 사용되는 언어입니다.	
Content-Length	본문의 크기(바이트)입니다.	
Content-Range	응답에서 반환된 개체 부분입니다.	
Content-Type	오브젝트 데이터의 형식을 설명하는 표준 MIME 유형.	
ETag	ETAG는 웹 서버가 URL에서 찾은 리소스의 특정 버전에 할당한 식별자입니다.	
Expires	개체를 더 이상 캐시할 수 없는 날짜 및 시간입니다.	
Last-Modified	개체를 만든 날짜입니다.	
x-amz-delete-marker	영구적으로 삭제된 버전이 지정된 개체가 삭제 마커인지(참) 또는 삭제 마커가 아닌지(거짓) 여부를 지정합니다.	
x-amz-expiration	개체 만료가 구성된 경우 응답에 이 헤더가 포함됩니다.	
x-amz-object-lock-mode	개체에 대해 현재 사용 중인 개체 잠금 모드입니다. Values : GOVERNANCE or COMPLIANCE	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
x-amz-server-side-encryption	Amazon S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘 Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	있는 경우 개체에 사용된 AWS KMS(AWS Key Management Service) 대칭 고객 관리 키의 ID를 지정합니다.	
x-amz-server-side-encryption-bucket-key-enabled	복사된 개체가 AWS KMS(SSE-KMS)와의 서버측 암호화에 S3 버킷 키를 사용하는지 여부를 나타냅니다.	
x-amz-server-side-encryption-customer-algorithm	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 사용된 암호화 알고리즘을 확인하는 헤더가 포함됩니다.	
x-amz-server-side-encryption-customer-key-MD5	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 이 헤더가 포함되어 고객이 제공한 암호화 키에 대한 왕복 메시지 무결성 확인을 제공합니다. 	
x-amz-storage-class	개체의 스토리지 클래스 정보를 제공합니다.	
x-amz-tagging-count	개체에 있는 태그 수(있는 경우)입니다.	
x-amz-version-id	개체의 버전 ID입니다.	
Body	Response Syntax 참조	

## Sample Response

```

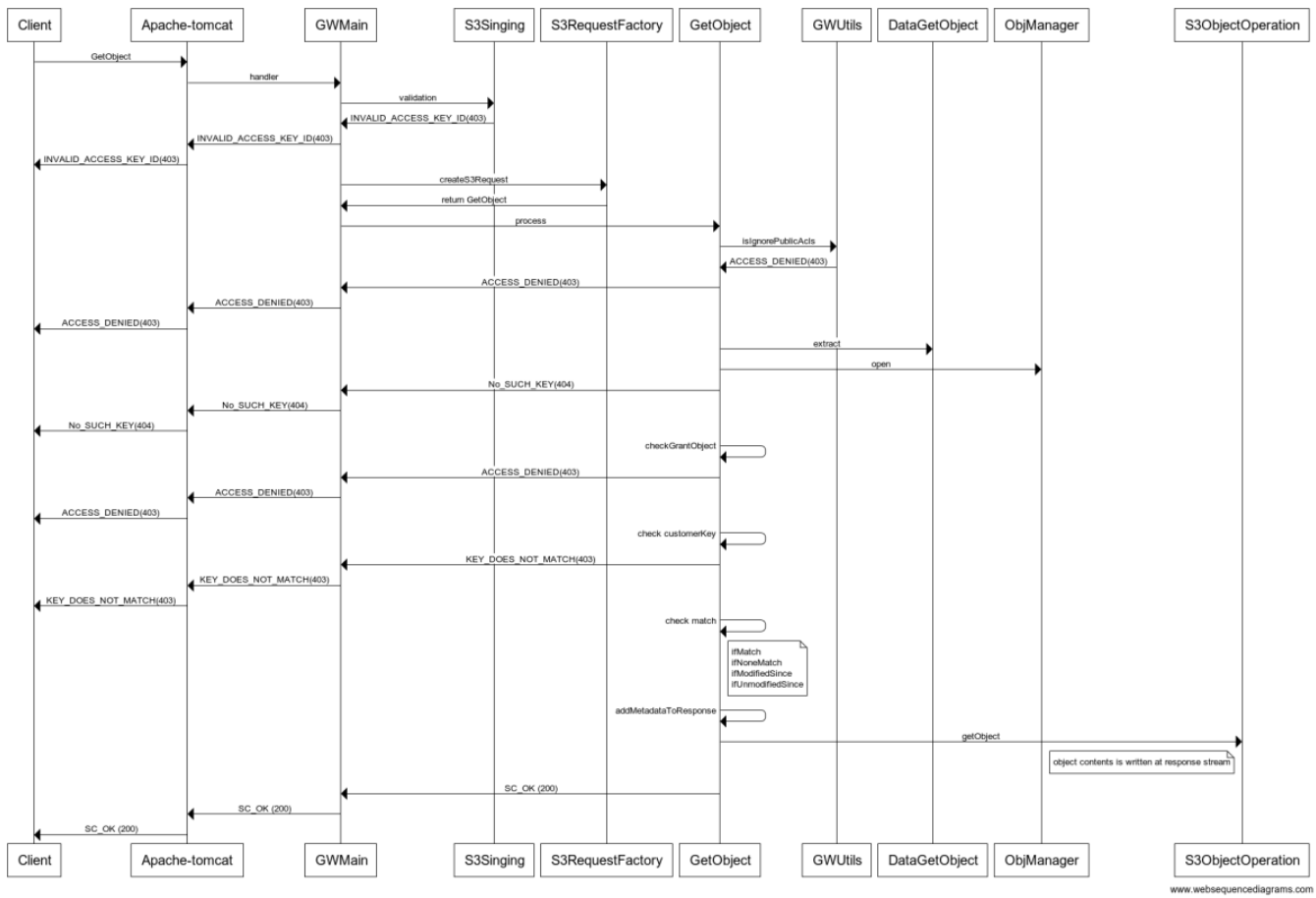
HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap540pIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
x-amz-version-id: 3/L4kqtJlcpXroDTDmJ+rmSpXd3QBpUMLUo
ETag: "fba9dede5f27731c9771645a39863328"
Content-Length: 434234
Content-Type: text/plain
Connection: close
Server: KSAN-GW
[434234 bytes of object data]
```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음

Status Code	Description
404 (No Suck Key)	Key가 존재하지 않음
500 (Server Error)	

# Processing of requests



# GetObjectAcl

## Overview

개체의 ACL(액세스 제어 목록)을 반환합니다.

## Request Syntax

```
GET /Bucket/Key?acl&versionId=VersionId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/my-image.jpg?acl HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
versionId	개체의 특정 버전을 참조하는 데 사용되는 VersionId입니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<AccessControlPolicy>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee>
        <DisplayName>string</DisplayName>
        <EmailAddress>string</EmailAddress>
        <ID>string</ID>
```

```

    <xsi:type>string</xsi:type>
    <URI>string</URI>
  </Grantee>
  <Permission>string</Permission>
</Grant>
</AccessControlList>
</AccessControlPolicy>

```

## Response Elements

Parameter Name	Description	Required
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
AccessControlPolicy	Response Syntax 참조	Yes

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51TnqcoF8eFidJG9Z/2mkiDFu8yU9AS1ed40pIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
x-amz-version-id: 4HL4kqtJlcpXroDTdmJ+rmSpXd3dIbrHY+MTRCxf3vjVBH40NrjfkD
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
Content-Length: 124
Content-Type: text/plain
Connection: close
Server: AmazonS3

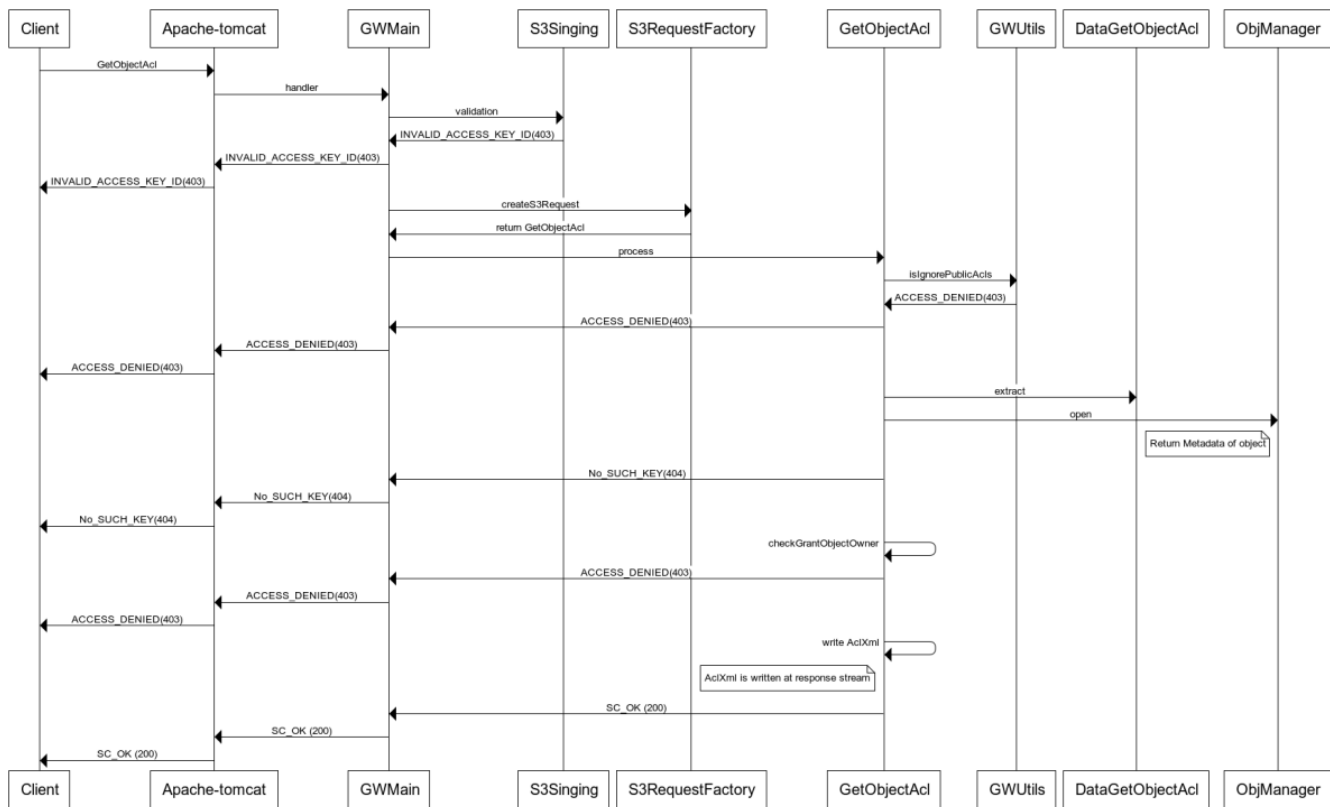
<AccessControlPolicy>
  <Owner>
    <ID>75aa57f09aa0c8caeb4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>mtD@amazon.com</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser">
        <ID>75aa57f09aa0c8caeb4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
        <DisplayName>mtD@amazon.com</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>

```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Suck Key)	Key가 존재하지 않음
500 (Server Error)	

## Processing of requests



# GetObjectRetention

## Overview

개체의 보존 설정을 표시합니다.

## Request Syntax

```
GET /Bucket/Key?retention&versionId=VersionId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
versionId	개체의 특정 버전을 참조하는 데 사용되는 VersionId입니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<Retention>
  <Mode>string</Mode>
  <RetainUntilDate>timestamp</RetainUntilDate>
</Retention>
```

## Response Elements

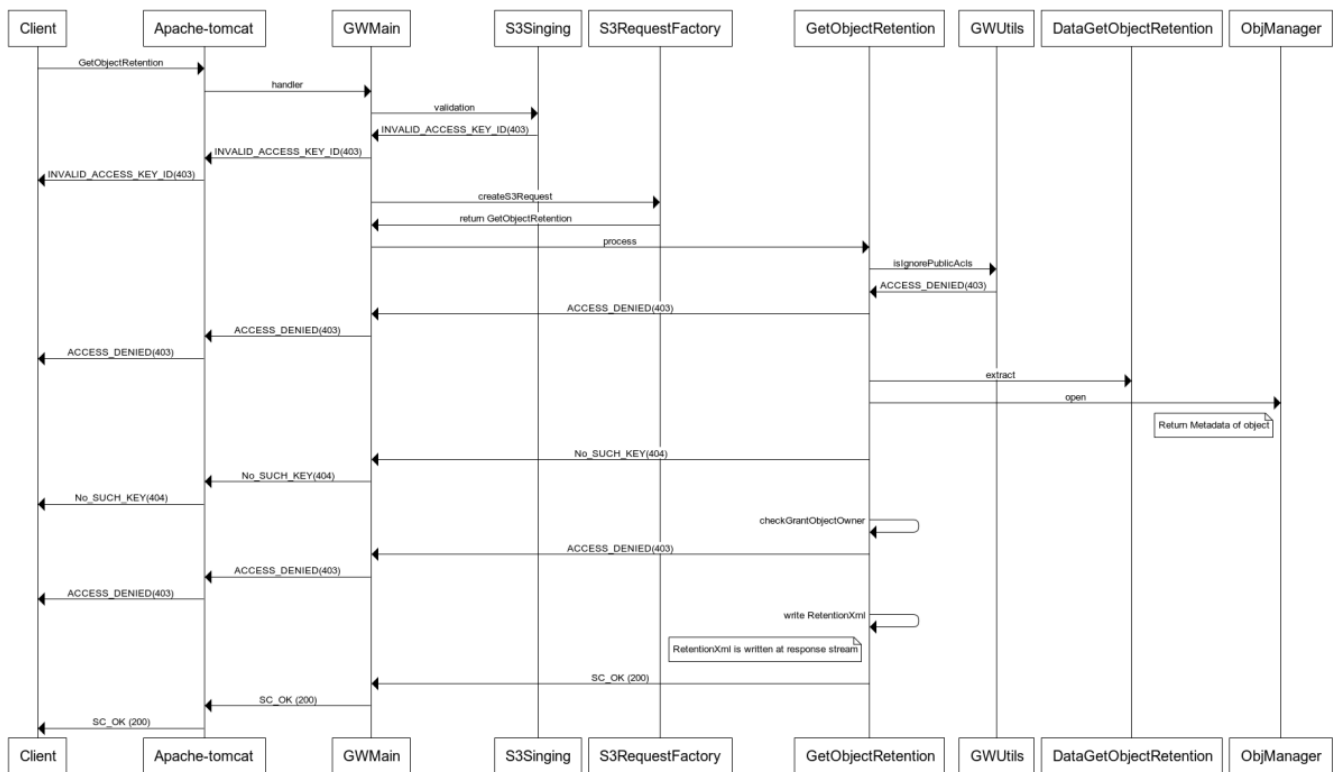
Parameter Name	Description	Required
Retention	REsponse Syntax 참조	Yes

# Sample Response

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Such Key)	Key가 존재하지 않음
500 (Server Error)	

## Processing of requests



# GetObjectTagging

## Overview

개체의 태그 집합을 반환합니다.

## Request Syntax

```
GET /Bucket/Key?tagging&versionId=VersionId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
x-amz-request-payer: RequestPayer
```

## Sample Request

```
GET /ExampleBucket/example-object?tagging HTTP/1.1
Host: 192.168.111.111:8080
Date: Thu, 22 Sep 2016 21:33:08 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
versionId	개체의 특정 버전을 참조하는 데 사용되는 VersionId입니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
x-amz-version-id: VersionId
<?xml version="1.0" encoding="UTF-8"?>
<Tagging>
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>
```



# Response Elements

Parameter Name	Description	Required
x-amz-version-id	태그 정보를 가져온 개체의 versionId입니다.	
Tagging	REsponse Syntax 참조	Yes

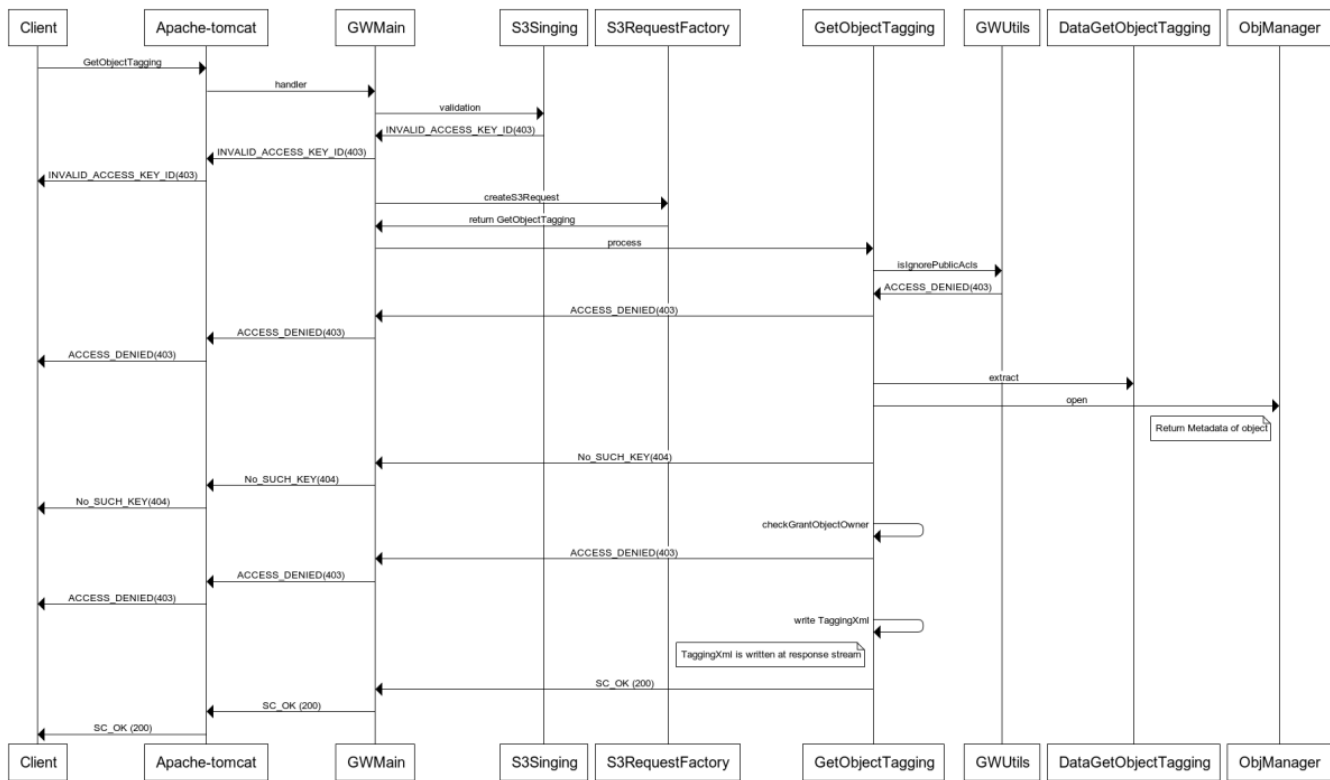
# Sample Response

```
HTTP/1.1 200 OK
Date: Thu, 22 Sep 2016 21:33:08 GMT
Connection: close
Server: AmazonS3
<?xml version="1.0" encoding="UTF-8"?>
<Tagging xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <TagSet>
    <Tag>
      <Key>tag1</Key>
      <Value>val1</Value>
    </Tag>
    <Tag>
      <Key>tag2</Key>
      <Value>val2</Value>
    </Tag>
  </TagSet>
</Tagging>
```

# HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Suck Key)	Key가 존재하지 않음
500 (Server Error)	

# Processing of requests



# GetPublicAccessBlock

## Overview

S3 버킷에 대한 PublicAccessBlock 구성을 검색합니다.

## Request Syntax

```
GET /Bucket/?publicAccessBlock HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?publicAccessBlock HTTP/1.1
Host: 192.168.111.111:8080
x-amz-date: <Thu, 15 Nov 2016 00:17:21 GMT>
Authorization: <signatureValue>
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<PublicAccessBlockConfiguration>
  <BlockPublicAcls>boolean</BlockPublicAcls>
  <IgnorePublicAcls>boolean</IgnorePublicAcls>
  <BlockPublicPolicy>boolean</BlockPublicPolicy>
  <RestrictPublicBuckets>boolean</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>
```

## Response Elements

Parameter Name	Description	Required
PublicAccessBlockConfiguration	REsponse Syntax 참조	Yes

## Sample Response

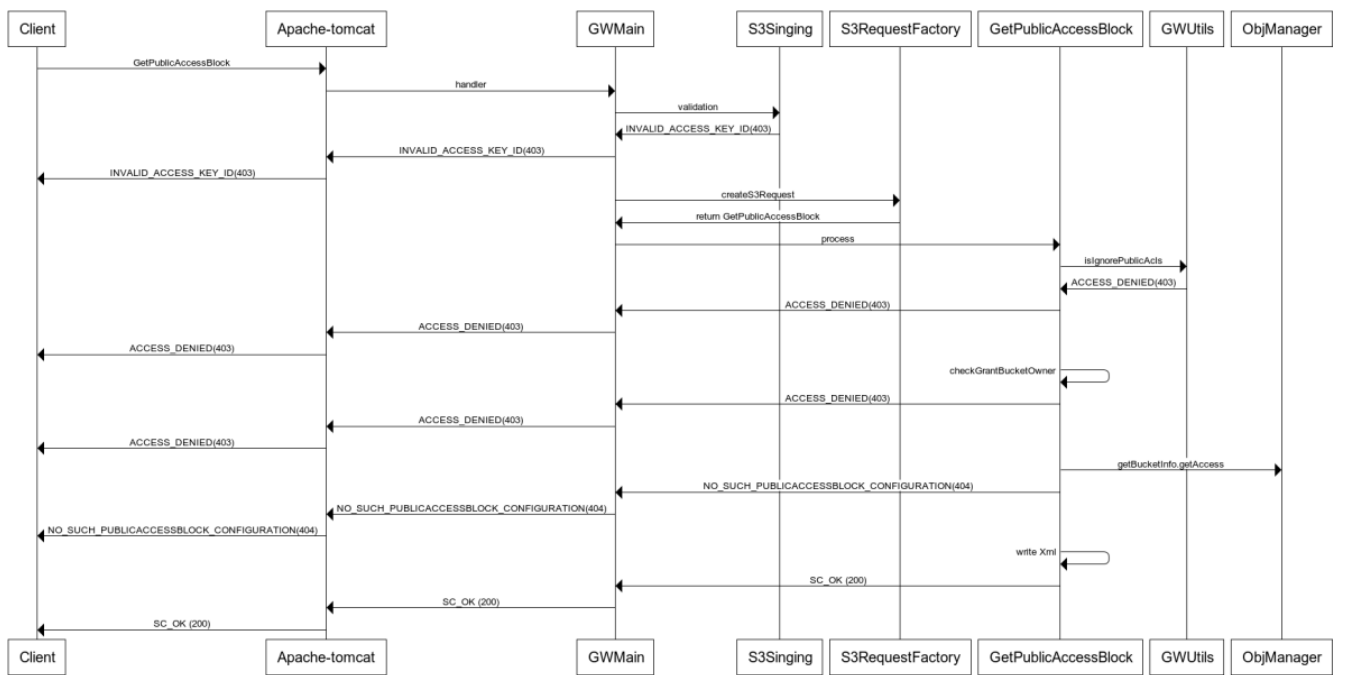
```
HTTP/1.1 200 OK
x-amz-id-2: ITnGT1y4REXAMPLEPi4hklTXouTf0hccUjo0iCPEXAMPLEutBj3M7fPGLW02SEWp
x-amz-request-id: 51991EXAMPLE5321
Date: Thu, 15 Nov 2016 00:17:22 GMT
Server: AmazonS3
Content-Length: 0
```

```
<PublicAccessBlockConfiguration>
  <BlockPublicAcls>TRUE</BlockPublicAcls>
  <IgnorePublicAcls>FALSE</IgnorePublicAcls>
  <BlockPublicPolicy>FALSE</BlockPublicPolicy>
  <RestrictPublicBuckets>FALSE</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>
```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# HeadBucket

## Overview

버킷이 있는지, 버킷에 액세스할 수 있는 권한이 있는지 확인하는 데 유용합니다.

## Request Syntax

```
HEAD /Bucket/ HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
HEAD /ExampleBucket/ HTTP/1.1
Date: Fri, 10 Feb 2012 21:34:55 GMT
Authorization: authorization string
Host: 192.168.111.111:8080
Connection: Keep-Alive
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

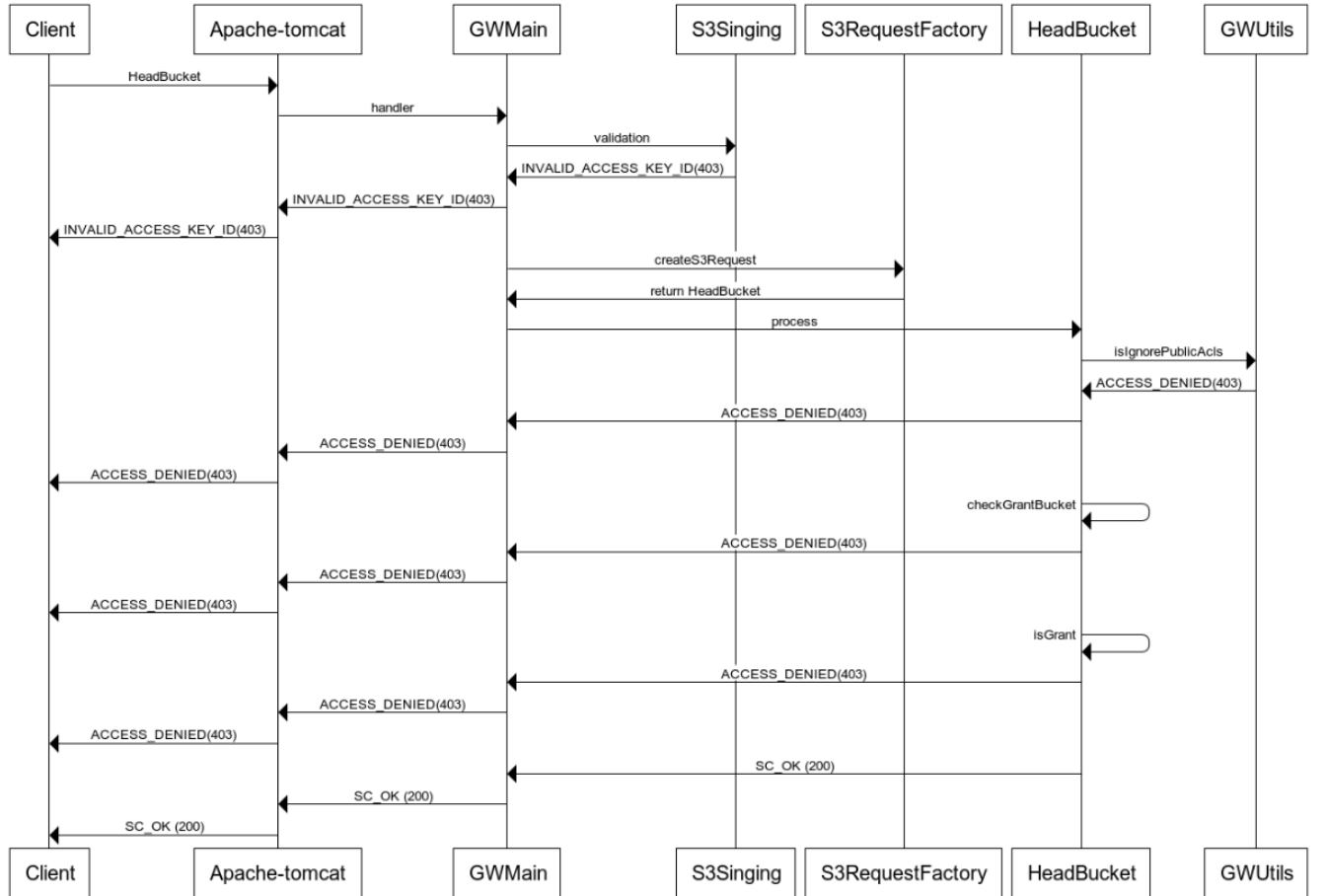
## Sample Response

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공

Status Code	Description
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# HeadObject

## Overview

개체에서 메타데이터를 검색합니다.

## Request Syntax



```
HEAD /Bucket/Key?partNumber=PartNumber&versionId=VersionId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
If-Match: IfMatch
If-Modified-Since: IfModifiedSince
If-None-Match: IfNoneMatch
If-Unmodified-Since: IfUnmodifiedSince
Range: Range
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
HEAD /ExampleBucket/my-image.jpg HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: AWS AKIAIOSFODNN7EXAMPLE:02236Q3V0RonhpaBX5sCYVf1bNRuU=
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
versionId	개체의 특정 버전을 참조하는 데 사용되는 VersionId입니다.	
If-Match	해당 엔티티 태그(ETAG)가 지정된 개체와 동일한 경우에만 개체를 반환하고 그렇지 않으면 412(전제 조건 실패)를 반환합니다.	
If-Modified-Since	지정된 시간 이후 개체를 수정한 경우에만 개체를 반환하고, 그렇지 않으면 304(수정되지 않음)를 반환합니다.	
If-None-Match	해당 엔티티 태그(ETAG)가 지정된 개체와 다른 경우에만 개체를 반환하고, 그렇지 않으면 수정되지 않은 304를 반환합니다.	
If-Unmodified-Since	지정된 시간 이후 개체를 수정하지 않은 경우에만 개체를 반환하고, 그렇지 않으면 412(전제 조건 실패)를 반환합니다.	
Range	개체의 지정된 범위 바이트를 다운로드합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	
x-amz-server-side-encryption-customer-algorithm	개체를 암호화할 때 사용할 알고리즘(예: AES256)을 지정합니다.	

Parameter Name	Description	Required
x-amz-server-side-encryption-customer-key	데이터 암호화에 사용할 S3에 대해 고객이 제공한 암호화 키를 지정합니다. 	
x-amz-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 128비트 MD5 다이제스트를 지정합니다. 	

## Request Body

## Response Syntax

```

HTTP/1.1 200
x-amz-delete-marker: DeleteMarker
accept-ranges: AcceptRanges
x-amz-expiration: Expiration
x-amz-restore: Restore
x-amz-archive-status: ArchiveStatus
Last-Modified: LastModified
Content-Length: ContentLength
ETag: ETag
x-amz-missing-meta: MissingMeta
x-amz-version-id: VersionId
Cache-Control: CacheControl
Content-Disposition: ContentDisposition
Content-Encoding: ContentEncoding
Content-Language: ContentLanguage
Content-Type: ContentType
Expires: Expires
x-amz-website-redirect-location: WebsiteRedirectLocation
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-storage-class: StorageClass
x-amz-request-charged: RequestCharged
x-amz-replication-status: ReplicationStatus
x-amz-mp-parts-count: PartsCount
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
x-amz-object-lock-legal-hold: ObjectLockLegalHoldStatus

```

## Response Elements

Parameter Name	Description	Required
accept-ranges	바이트 범위가 지정되었음을 나타냅니다.	
Cache-Control	요청/응답 체인을 따라 캐싱 동작을 지정합니다.	
Content-Disposition	개체에 대한 프레젠테이션 정보를 지정합니다.	
Content-Encoding	개체에 적용된 콘텐츠 인코딩과 콘텐츠 유형 헤더 필드가 참조하는 미디어 유형을 가져오기 위해 적용해야 하는 디코딩 메커니즘을 지정합니다.	
Content-Language	콘텐츠가 사용되는 언어입니다.	
Content-Length	본문의 크기(바이트)입니다.	
Content-Range	응답에서 반환된 개체 부분입니다.	
Content-Type	오브젝트 데이터의 형식을 설명하는 표준 MIME 유형.	
ETag	ETAG는 웹 서버가 URL에서 찾은 리소스의 특정 버전에 할당한 식별자입니다.	



Parameter Name	Description	Required
Expires	개체를 더 이상 캐시할 수 없는 날짜 및 시간입니다.	
Last-Modified	개체를 만든 날짜입니다.	
x-amz-delete-marker	영구적으로 삭제된 버전이 지정된 개체가 삭제 마커인지 (참) 또는 삭제 마커가 아닌지(거짓) 여부를 지정합니다.	
x-amz-expiration	개체 만료가 구성된 경우 응답에 이 헤더가 포함됩니다.	
x-amz-object-lock-mode	개체에 대해 현재 사용 중인 개체 잠금 모드입니다. Values : GOVERNANCE or COMPLIANCE	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
x-amz-server-side-encryption	Amazon S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘 Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	있는 경우 개체에 사용된 AWS KMS(AWS Key Management Service) 대칭 고객 관리 키의 ID를 지정합니다.	
x-amz-server-side-encryption-bucket-key-enabled	복사된 개체가 AWS KMS(SSE-KMS)와의 서버측 암호화에 S3 버킷 키를 사용하는지 여부를 나타냅니다.	
x-amz-server-side-encryption-customer-algorithm	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 사용된 암호화 알고리즘을 확인하는 헤더가 포함됩니다.	
x-amz-server-side-encryption-customer-key-MD5	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 이 헤더가 포함되어 고객이 제공한 암호화 키에 대한 왕복 메시지 무결성 확인을 제공합니다. 	
x-amz-storage-class	개체의 스토리지 클래스 정보를 제공합니다.	
x-amz-tagging-count	개체에 있는 태그 수(있는 경우)입니다.	
x-amz-version-id	개체의 버전 ID입니다.	

## Sample Response

```

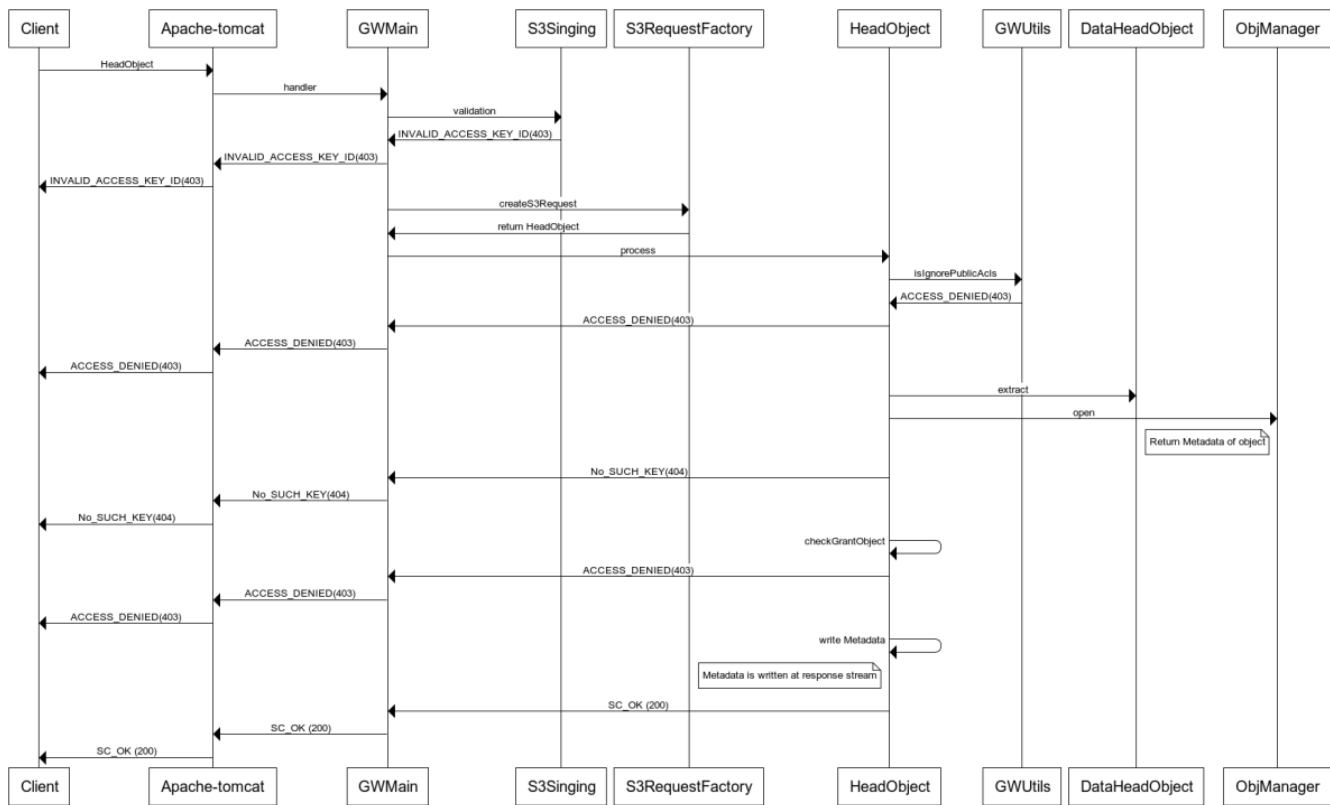
HTTP/1.1 200 OK
x-amz-id-2: ef8yU9AS1ed40pIszj7UDNEHGran
x-amz-request-id: 318BC8BC143432E5
x-amz-version-id: 3HL4kqtJlcpXroDTDmjVBH40NrjfkD
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
ETag: "fba9dede5f27731c9771645a39863328"
Content-Length: 434234
Content-Type: text/plain
Connection: close
Server: KSAN-GW

```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Suck Key)	Key가 존재하지 않음
500 (Server Error)	

## Processing of requests



# ListBuckets

## Overview

인증된 사용자가 소유한 모든 버킷 목록을 반환합니다.

## Request Syntax

GET / HTTP/1.1

## Sample Request

## URI request Parameters

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListAllMyBucketsResult>
  <Buckets>
    <Bucket>
      <CreationDate>timestamp</CreationDate>
      <Name>string</Name>
    </Bucket>
  </Buckets>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
</ListAllMyBucketsResult>
```

## Response Elements

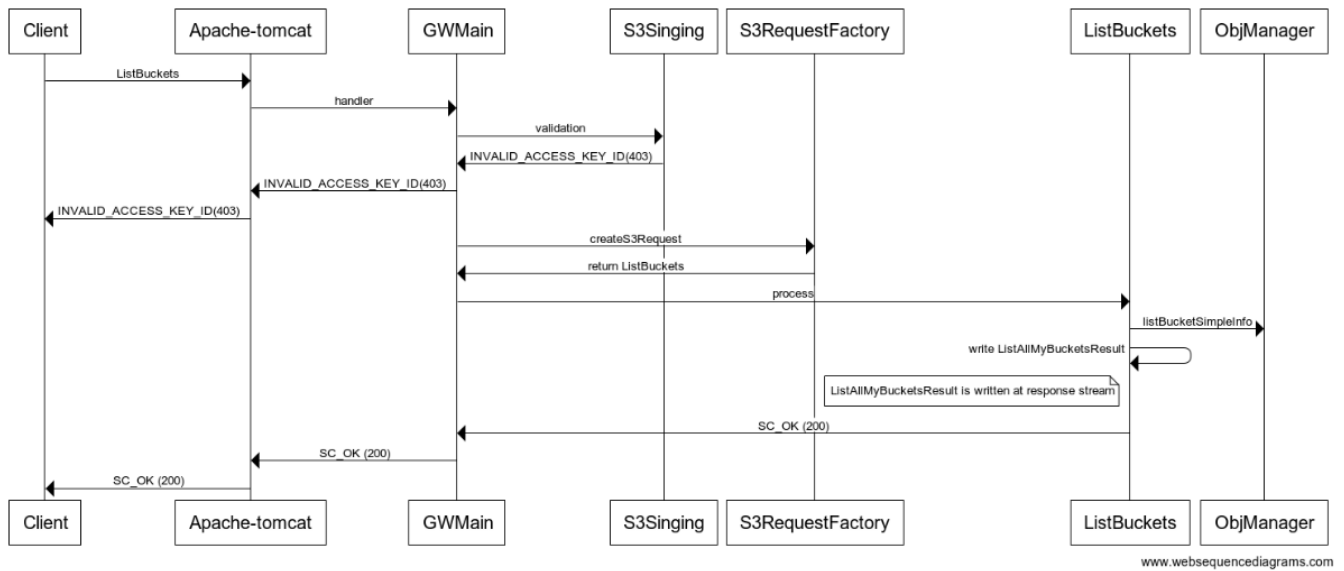
Parameter Name	Description	Required
ListAllMyBucketsResult	RResponse Syntax 참조	Yes

## Sample Response

# HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
500 (Server Error)	

## Processing of requests



# ListMultipartUploads

## Overview

진행 중인 다중 파트 업로드를 나열합니다.

## Request Syntax

```
GET /Bucket/?uploads&delimiter=Delimiter&encoding-type=EncodingType&key-marker=KeyMarker&max-uploads=MaxUploads&prefix=Prefix&upload-id-marker=UploadIdMarker HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/?uploads&max-uploads=3 HTTP/1.1
Host: 192.168.111.111:8080
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
delimiter	Key를 그룹화하는 데 사용하는 문자입니다.	
encoding-type	S3에 응답의 개체 키를 인코딩하도록 요청하고 사용할 인코딩 방법을 지정합니다. Values : url	
key-marker	upload-id-marker와 함께 목록을 시작할 다중 파트 업로드를 지정합니다.	
max-uploads	응답에서 반환할 수 있는 최대 업로드 수입니다. 최대 1,000개	
prefix	지정된 접두사로 시작하는 키에 대해서만 진행 중인 업로드를 나열합니다.	
upload-id-marker	키 표시기와 함께 목록을 시작할 다중 파트 업로드를 지정합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListMultipartUploadsResult>
  <Bucket>string</Bucket>
  <KeyMarker>string</KeyMarker>
```

```

<UploadIdMarker>string</UploadIdMarker>
<NextKeyMarker>string</NextKeyMarker>
<Prefix>string</Prefix>
<Delimiter>string</Delimiter>
<NextUploadIdMarker>string</NextUploadIdMarker>
<MaxUploads>integer</MaxUploads>
<IsTruncated>boolean</IsTruncated>
<Upload>
  <Initiated>timestamp</Initiated>
  <Initiator>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Initiator>
  <Key>string</Key>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
  <StorageClass>string</StorageClass>
  <UploadId>string</UploadId>
</Upload>
...
<CommonPrefixes>
  <Prefix>string</Prefix>
</CommonPrefixes>
...
<EncodingType>string</EncodingType>
</ListMultipartUploadsResult>

```

## Response Elements

Parameter Name	Description	Required
ListMultipartUploadsResult	REsponse Syntax 참조	Yes

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: Uuag1LuByRx9e6j50nimru9p04ZVKnJ2Qz7/C1NPcfTWAtRPfTa0Fg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
Content-Length: 1330
Connection: keep-alive
Server: KSAN-GW

<?xml version="1.0" encoding="UTF-8"?>
<ListMultipartUploadsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Bucket>bucket</Bucket>
  <KeyMarker></KeyMarker>
  <UploadIdMarker></UploadIdMarker>
  <NextKeyMarker>my-movie.m2ts</NextKeyMarker>
  <NextUploadIdMarker>YW55IGlkZWEd2h5IGVsdmluZydzIHVwbG9hZCBmYWlsZWQ</NextUploadIdMarker>
  <MaxUploads>3</MaxUploads>
  <IsTruncated>true</IsTruncated>
  <Upload>
    <Key>my-divisor</Key>
    <UploadId>XMgbGlrZSB1bHZpbmcncyBub3QgaGF2aW5nIG11Y2ggbHVjaw</UploadId>
    <Initiator>
      <ID>arn:aws:iam::111122223333:user/user1-11111a31-17b5-4fb7-9df5-b11111f13de</ID>
      <DisplayName>user1-11111a31-17b5-4fb7-9df5-b11111f13de</DisplayName>
    </Initiator>
    <Owner>
      <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
      <DisplayName>OwnerDisplayName</DisplayName>
    </Owner>
  </Upload>

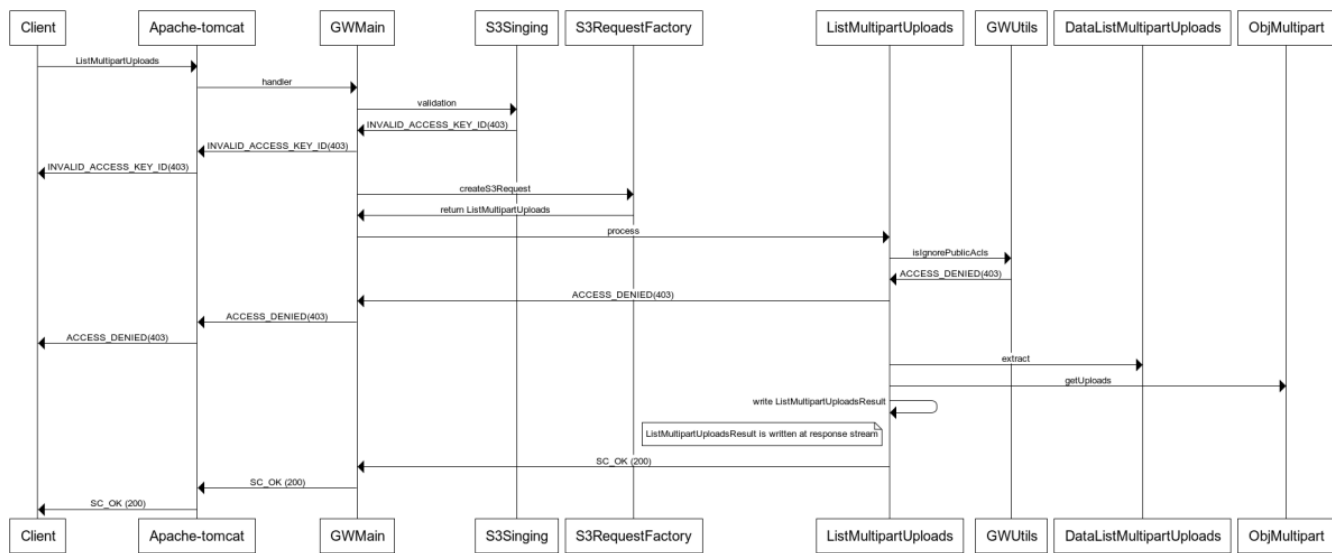
```

```
</Owner>
<StorageClass>STANDARD</StorageClass>
<Initiated>2010-11-10T20:48:33.000Z</Initiated>
</Upload>
<Upload>
  <Key>my-movie.m2ts</Key>
  <UploadId>VXBsb2FkIElEIGZvcjBlbHZpbmcncyBteS1tb3ZpZS5tMnRzIHVwbG9hZA</UploadId>
  <Initiator>
    <ID>b1d16700c70b0b05597d7acd6a3f92be</ID>
    <DisplayName>InitiatorDisplayName</DisplayName>
  </Initiator>
  <Owner>
    <ID>b1d16700c70b0b05597d7acd6a3f92be</ID>
    <DisplayName>OwnerDisplayName</DisplayName>
  </Owner>
  <StorageClass>STANDARD</StorageClass>
  <Initiated>2010-11-10T20:48:33.000Z</Initiated>
</Upload>
<Upload>
  <Key>my-movie.m2ts</Key>
  <UploadId>YW55IGlkZWEd2h5IGVsdmluZydzIHVwbG9hZCBmYWlsZWQ</UploadId>
  <Initiator>
    <ID>arn:aws:iam::444455556666:user/user1-2222a31-17b5-4fb7-9df5-b22222f13de</ID>
    <DisplayName>user1-2222a31-17b5-4fb7-9df5-b22222f13de</DisplayName>
  </Initiator>
  <Owner>
    <ID>b1d16700c70b0b05597d7acd6a3f92be</ID>
    <DisplayName>OwnerDisplayName</DisplayName>
  </Owner>
  <StorageClass>STANDARD</StorageClass>
  <Initiated>2010-11-10T20:49:33.000Z</Initiated>
</Upload>
</ListMultipartUploadsResult>
```

# HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Suck uploadId)	uploadId가 존재하지 않음
500 (Server Error)	

# Processing of requests





# ListObjects

## Overview

버킷에 있는 개체의 일부 또는 전체(최대 1,000개)를 반환합니다.

## Request Syntax

```
GET /Bucket/?delimiter=Delimiter&encoding-type=EncodingType&marker=Marker&max-keys=MaxKeys&prefix=Prefix HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/ HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
delimiter	Key를 그룹화하는 데 사용하는 문자입니다.	
encoding-type	S3에 응답의 개체 키를 인코딩하도록 요청하고 사용할 인코딩 방법을 지정합니다. Values : url	
marker	S3가 리스팅을 시작하길 원하는 곳입니다.	
max-keys	응답에 반환되는 키의 최대 수를 설정합니다. 기본적으로 이 동작은 최대 1,000개의 키 이름을 반환합니다.	
prefix	지정된 접두사로 시작하는 키에 대한 응답을 제한합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult>
  <IsTruncated>boolean</IsTruncated>
  <Marker>string</Marker>
  <NextMarker>string</NextMarker>
  <Contents>
```

```

    <ETag>string</ETag>
    <Key>string</Key>
    <LastModified>timestamp</LastModified>
    <Owner>
        <DisplayName>string</DisplayName>
        <ID>string</ID>
    </Owner>
    <Size>integer</Size>
    <StorageClass>string</StorageClass>
</Contents>
...
<Name>string</Name>
<Prefix>string</Prefix>
<Delimiter>string</Delimiter>
<MaxKeys>integer</MaxKeys>
<CommonPrefixes>
    <Prefix>string</Prefix>
</CommonPrefixes>
...
<EncodingType>string</EncodingType>
</ListBucketResult>

```

## Response Elements

Parameter Name	Description	Required
ListBucketResult	RResponse Syntax 참조	Yes

## Sample Response

```

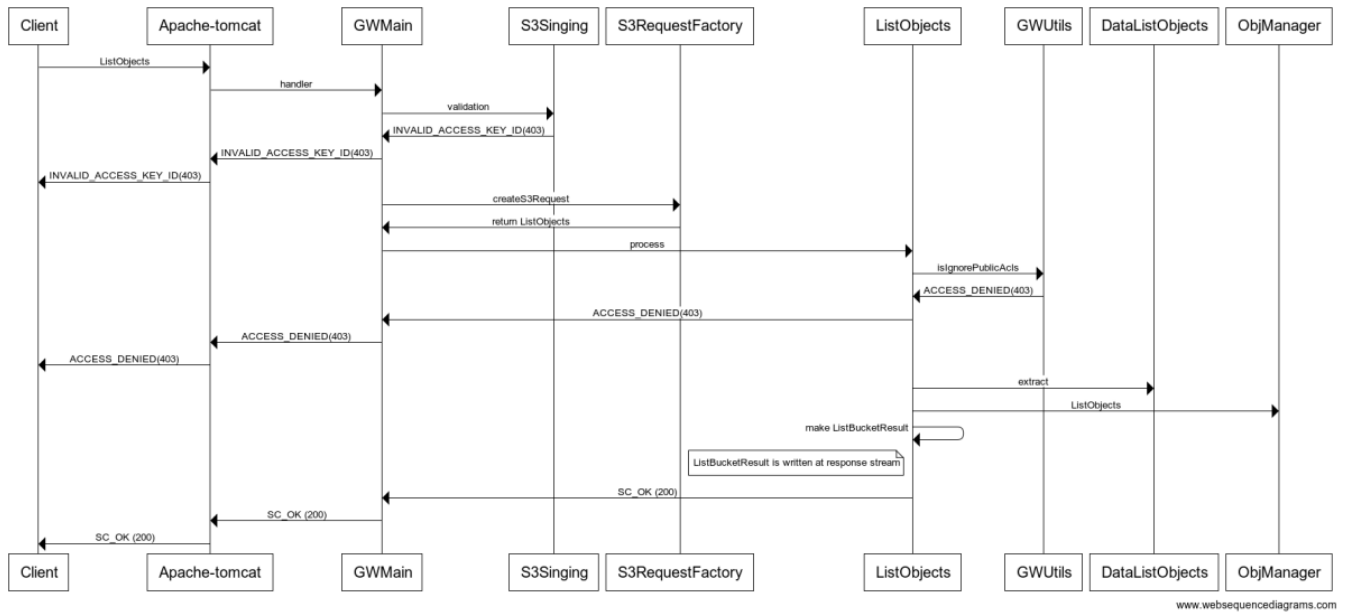
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
    <Name>bucket</Name>
    <Prefix/>
    <Marker/>
    <MaxKeys>1000</MaxKeys>
    <IsTruncated>false</IsTruncated>
    <Contents>
        <Key>my-image.jpg</Key>
        <LastModified>2009-10-12T17:50:30.000Z</LastModified>
        <ETag>"fba9dede5f27731c9771645a39863328"</ETag>
        <Size>434234</Size>
        <StorageClass>STANDARD</StorageClass>
        <Owner>
            <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
            <DisplayName>mtd@amazon.com</DisplayName>
        </Owner>
    </Contents>
    <Contents>
        <Key>my-third-image.jpg</Key>
        <LastModified>2009-10-12T17:50:30.000Z</LastModified>
        <ETag>"1b2cf535f27731c974343645a3985328"</ETag>
        <Size>64994</Size>
        <StorageClass>STANDARD_IA</StorageClass>
        <Owner>
            <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
            <DisplayName>mtd@amazon.com</DisplayName>
        </Owner>
    </Contents>
</ListBucketResult>

```

# HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# ListObjectsV2

## Overview

각 요청과 함께 버킷에 있는 개체의 일부 또는 전체(최대 1,000개)를 반환합니다.

## Request Syntax

```
GET /Bucket/?list-type=2&continuation-token=ContinuationToken&delimiter=Delimiter&encoding-type=EncodingType&fetch-owner=FetchOwner&max-keys=MaxKeys&prefix=Prefix&start-after=StartAfter HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
continuation-token	버킷에서 토큰과 함께 목록이 계속되고 있음을 나타냅니다.	
delimiter	Key를 그룹화하는 데 사용하는 문자입니다.	
encoding-type	S3에 응답의 개체 키를 인코딩하도록 요청하고 사용할 인코딩 방법을 지정합니다. Values : url	
fetch-owner	owner 필드는 listV2에 기본적으로 없습니다.	
max-keys	응답에 반환되는 키의 최대 수를 설정합니다. 기본적으로 이 동작은 최대 1,000개의 키 이름을 반환합니다.	
prefix	지정된 접두사로 시작하는 키에 대한 응답을 제한합니다.	
start-after	S3가 리스팅을 시작할길 원하는 곳입니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult>
  <IsTruncated>boolean</IsTruncated>
  <Contents>
    <ETag>string</ETag>
```

```

    <Key>string</Key>
    <LastModified>timestamp</LastModified>
    <Owner>
      <DisplayName>string</DisplayName>
      <ID>string</ID>
    </Owner>
    <Size>integer</Size>
    <StorageClass>string</StorageClass>
  </Contents>
  ...
  <Name>string</Name>
  <Prefix>string</Prefix>
  <Delimiter>string</Delimiter>
  <MaxKeys>integer</MaxKeys>
  <CommonPrefixes>
    <Prefix>string</Prefix>
  </CommonPrefixes>
  ...
  <EncodingType>string</EncodingType>
  <KeyCount>integer</KeyCount>
  <ContinuationToken>string</ContinuationToken>
  <NextContinuationToken>string</NextContinuationToken>
  <StartAfter>string</StartAfter>
</ListBucketResult>

```

## Response Elements

Parameter Name	Description	Required
ListBucketResult	RResponse Syntax 참조	Yes

## Sample Response

```

<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>bucket</Name>
  <Prefix/>
  <KeyCount>205</KeyCount>
  <MaxKeys>1000</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <Contents>
    <Key>my-image.jpg</Key>
    <LastModified>2009-10-12T17:50:30.000Z</LastModified>
    <ETag>"fba9dede5f27731c9771645a39863328"</ETag>
    <Size>434234</Size>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  <Contents>
    ...
  </Contents>
  ...
</ListBucketResult>

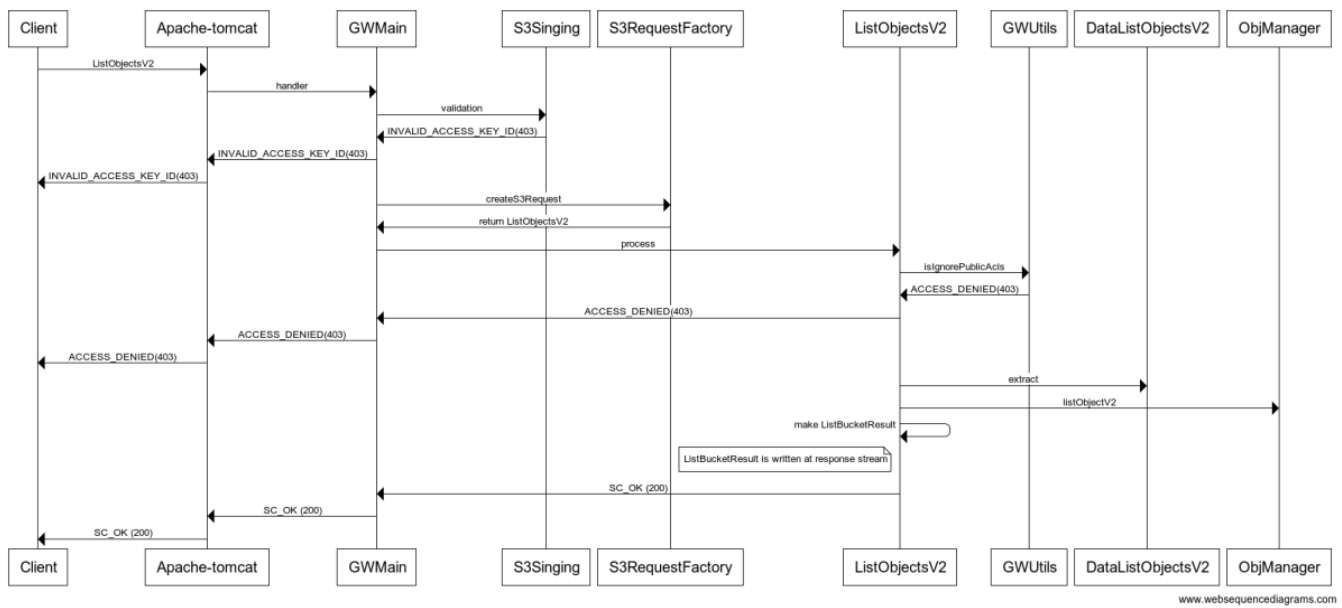
```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음

Status Code	Description
500 (Server Error)	

# Processing of requests



# ListObjectVersions

## Overview

버킷에 있는 개체의 모든 버전에 대한 메타데이터를 반환합니다.

## Request Syntax

```
GET /Bucket/?versions&delimiter=Delimiter&encoding-type=EncodingType&key-marker=KeyMarker&max-keys=MaxKeys&prefix=Prefix&version-id-marker=VersionIdMarker HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /?versions HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 28 Oct 2009 22:32:00 +0000
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
delimiter	Key를 그룹화하는 데 사용하는 문자입니다.	
encoding-type	S3에 응답의 개체 키를 인코딩하도록 요청하고 사용할 인코딩 방법을 지정합니다. Values : url 	
key-marker	upload-id-marker와 함께 목록을 시작할 다중 파트 업로드를 지정합니다.	
max-keys	응답에 반환되는 키의 최대 수를 설정합니다.	
prefix	지정된 접두사로 시작하는 키만 선택하려면 이 매개 변수를 사용합니다.	
upload-id-marker	키 표시기와 함께 목록을 시작할 다중 파트 업로드를 지정합니다. 	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
<?xml version="1.0" encoding="UTF-8"?>
<ListVersionsResult>
  <IsTruncated>boolean</IsTruncated>
  <KeyMarker>string</KeyMarker>
  <VersionIdMarker>string</VersionIdMarker>
```

```

<NextKeyMarker>string</NextKeyMarker>
<NextVersionIdMarker>string</NextVersionIdMarker>
<Version>
  <ETag>string</ETag>
  <IsLatest>boolean</IsLatest>
  <Key>string</Key>
  <LastModified>timestamp</LastModified>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
  <Size>integer</Size>
  <StorageClass>string</StorageClass>
  <VersionId>string</VersionId>
</Version>
...
<DeleteMarker>
  <IsLatest>boolean</IsLatest>
  <Key>string</Key>
  <LastModified>timestamp</LastModified>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
  <VersionId>string</VersionId>
</DeleteMarker>
...
<Name>string</Name>
<Prefix>string</Prefix>
<Delimiter>string</Delimiter>
<MaxKeys>integer</MaxKeys>
<CommonPrefixes>
  <Prefix>string</Prefix>
</CommonPrefixes>
...
<EncodingType>string</EncodingType>
</ListVersionsResult>

```

## Response Elements

Parameter Name	Description	Required
ListVersionsResult	RResponse Syntax 참조	Yes

## Sample Response

```

<?xml version="1.0" encoding="UTF-8"?>

<ListVersionsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01">
  <Name>bucket</Name>
  <Prefix>my</Prefix>
  <KeyMarker/>
  <VersionIdMarker/>
  <MaxKeys>5</MaxKeys>
  <IsTruncated>false</IsTruncated>
  <Version>
    <Key>my-image.jpg</Key>
    <VersionId>3/L4kqtJL40Nr8X8gdRQBpUMLUo</VersionId>
    <IsLatest>true</IsLatest>
    <LastModified>2009-10-12T17:50:30.000Z</LastModified>
    <ETag>"fba9dede5f27731c9771645a39863328"</ETag>
    <Size>434234</Size>
    <StorageClass>STANDARD</StorageClass>
    <Owner>

```



```

<ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
<DisplayName>mtd@amazon.com</DisplayName>
</Owner>
</Version>
<DeleteMarker>
  <Key>my-second-image.jpg</Key>
  <VersionId>03jpff543dhffds434rfdFDN943fdsFkdmqnh892</VersionId>
  <IsLatest>true</IsLatest>
  <LastModified>2009-11-12T17:50:30.000Z</LastModified>
  <Owner>
    <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>mtd@amazon.com</DisplayName>
  </Owner>
</DeleteMarker>
<Version>
  <Key>my-second-image.jpg</Key>
  <VersionId>QUpfndndhfd8438MNFNDN93jdnJFkdmqnh893</VersionId>
  <IsLatest>false</IsLatest>
  <LastModified>2009-10-10T17:50:30.000Z</LastModified>
  <ETag>"9b2cf535f27731c974343645a3985328"</ETag>
  <Size>166434</Size>
  <StorageClass>STANDARD</StorageClass>
  <Owner>
    <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>mtd@amazon.com</DisplayName>
  </Owner>
</Version>
<DeleteMarker>
  <Key>my-third-image.jpg</Key>
  <VersionId>03jpff543dhffds434rfdFDN943fdsFkdmqnh892</VersionId>
  <IsLatest>true</IsLatest>
  <LastModified>2009-10-15T17:50:30.000Z</LastModified>
  <Owner>
    <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>mtd@amazon.com</DisplayName>
  </Owner>
</DeleteMarker>
<Version>
  <Key>my-third-image.jpg</Key>
  <VersionId>UI0RUUnfdfhnw89493jJFJ</VersionId>
  <IsLatest>false</IsLatest>
  <LastModified>2009-10-11T12:50:30.000Z</LastModified>
  <ETag>"772cf535f27731c974343645a3985328"</ETag>
  <Size>64</Size>
  <StorageClass>STANDARD</StorageClass>
  <Owner>
    <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>mtd@amazon.com</DisplayName>
  </Owner>
</Version>
</ListVersionsResult>

```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# ListParts

## Overview

특정 멀티파트 업로드를 위해 업로드된 부분을 나열합니다.

## Request Syntax

```
GET /Bucket/Key?max-parts=MaxParts&part-number-marker=PartNumberMarker&uploadId=UploadId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

## Sample Request

```
GET /ExampleBucket/example-object?uploadId=XXBsb2FkIElEIGZvciBlbHZpbmcncyVcdS1tb3ZpZS5tMnRzEEEwbG9hZA&max-
parts=2&part-number-marker=1 HTTP/1.1
Host: 192.168.111.111:8080
Date: Mon, 1 Nov 2010 20:34:56 GMT
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	대상 개체 명	Yes
max-parts	반환할 최대 part 수를 설정합니다.	
part-number-marker	목록을 시작해야 하는 부분을 지정합니다. 번호가 더 높은 part만 나열됩니다. 	
uploadId	해당 part가 나열되는 멀티파트 업로드를 식별하는 upload ID입니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(엑세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

## Request Body

## Response Syntax


```
HTTP/1.1 200
x-amz-abort-date: AbortDate
x-amz-abort-rule-id: AbortRuleId
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<ListPartsResult>
  <Bucket>string</Bucket>
  <Key>string</Key>
  <UploadId>string</UploadId>
```

```

<PartNumberMarker>integer</PartNumberMarker>
<NextPartNumberMarker>integer</NextPartNumberMarker>
<MaxParts>integer</MaxParts>
<IsTruncated>boolean</IsTruncated>
<Part>
  <ETag>string</ETag>
  <LastModified>timestamp</LastModified>
  <PartNumber>integer</PartNumber>
  <Size>integer</Size>
</Part>
...
<Initiator>
  <DisplayName>string</DisplayName>
  <ID>string</ID>
</Initiator>
<Owner>
  <DisplayName>string</DisplayName>
  <ID>string</ID>
</Owner>
<StorageClass>string</StorageClass>
</ListPartsResult>

```

## Response Elements

Parameter Name	Description	Required
x-amz-abort-date	버킷에 불완전한 다중 파트 업로드를 중단하는 작업으로 구성된 수명 주기 규칙이 있고 수명 주기 규칙의 접두사가 요청의 개체 이름과 일치하면 시작된 다중 파트 업로드를 중단 작업에 적합한 시기를 나타내는 이 헤더가 응답에 포함됩니다. 	
x-amz-abort-rule-id	x-amz-abort-date 헤더와 함께 반환됩니다. 불완전한 다중 파트 업로드를 중단하는 작업을 정의하는 적용 가능한 라이프사이클 구성 규칙을 식별합니다.	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
ListPartsResult	Response Syntax 참조	

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: Uuag1LuByRx9e6j50nimru9p04ZVKnJ2Qz7/C1NPcfTWAtRPfTa0Fg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
Content-Length: 985
Connection: keep-alive
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<ListPartsResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Bucket>example-bucket</Bucket>
  <Key>example-object</Key>
  <UploadId>XXBsb2FkIElEIGZvciBlbHZpbmcncyVcdS1tb3ZpZS5tMnRzEEEwbG9hZA</UploadId>
  <Initiator>
    <ID>arn:aws:iam::111122223333:user/some-user-11116a31-17b5-4fb7-9df5-b288870f11xx</ID>
    <DisplayName>umat-user-11116a31-17b5-4fb7-9df5-b288870f11xx</DisplayName>
  </Initiator>
  <Owner>
    <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>someName</DisplayName>
  </Owner>
  <StorageClass>STANDARD</StorageClass>
  <PartNumberMarker>1</PartNumberMarker>

```

```

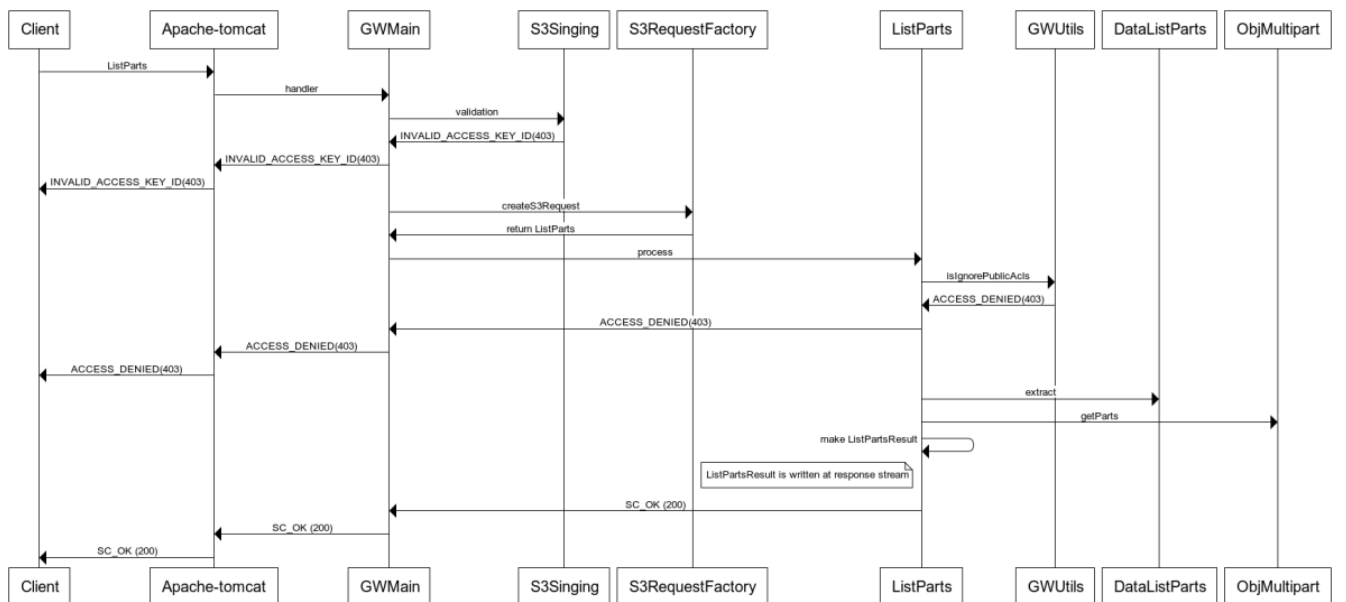
<NextPartNumberMarker>3</NextPartNumberMarker>
<MaxParts>2</MaxParts>
<IsTruncated>true</IsTruncated>
<Part>
  <PartNumber>2</PartNumber>
  <LastModified>2010-11-10T20:48:34.000Z</LastModified>
  <ETag>"7778aef83f66abc1fa1e8477f296d394"</ETag>
  <Size>10485760</Size>
</Part>
<Part>
  <PartNumber>3</PartNumber>
  <LastModified>2010-11-10T20:48:33.000Z</LastModified>
  <ETag>"aaaa18db4cc2f85cedef654fccc4a4x8"</ETag>
  <Size>10485760</Size>
</Part>
</ListPartsResult>

```

## HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Such uploadId)	uploadId가 존재하지 않음
500 (Server Error)	

## Processing of requests



# PutBucketAcl

## Overview

ACL(엑세스 제어 목록)을 사용하여 기존 버킷에 대한 권한을 설정합니다.

## Request Syntax

```
PUT /Bucket/?acl HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-acl: ACL
Content-MD5: ContentMD5
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write: GrantWrite
x-amz-grant-write-acp: GrantWriteACP
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<AccessControlPolicy xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <AccessControlList>
    <Grant>
      <Grantee>
        <DisplayName>string</DisplayName>
        <EmailAddress>string</EmailAddress>
        <ID>string</ID>
        <xsi:type>string</xsi:type>
        <URI>string</URI>
      </Grantee>
      <Permission>string</Permission>
    </Grant>
  </AccessControlList>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
</AccessControlPolicy>
```

## Sample Request

```
PUT /ExampleBucket/?acl HTTP/1.1
Host: 192.168.111.111:8080
Content-Length: 1660
x-amz-date: Thu, 12 Apr 2012 20:04:21 GMT
Authorization: authorization string

<AccessControlPolicy xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Owner>
    <ID>852b113e7a2f25102679df27bb0ae12b3f85be6BucketOwnerCanonicalUserID</ID>
    <DisplayName>OwnerDisplayName</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser">
        <ID>852b113e7a2f25102679df27bb0ae12b3f85be6BucketOwnerCanonicalUserID</ID>
        <DisplayName>OwnerDisplayName</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
    </Grant>
```

```

<Grant>
  <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="Group">
    <URI xmlns="">http://acs.amazonaws.com/groups/global/AllUsers</URI>
  </Grantee>
  <Permission xmlns="">READ</Permission>
</Grant>
<Grant>
  <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="Group">
    <URI xmlns="">http://acs.amazonaws.com/groups/s3/LogDelivery</URI>
  </Grantee>
  <Permission xmlns="">WRITE</Permission>
</Grant>
<Grant>
  <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="AmazonCustomerByEmail">
    <EmailAddress xmlns="">xyz@amazon.com</EmailAddress>
  </Grantee>
  <Permission xmlns="">WRITE_ACP</Permission>
</Grant>
<Grant>
  <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser">
    <ID xmlns="">f30716ab7115dcb44a5ef76e9d74b8e20567f63TestAccountCanonicalUserID</ID>
  </Grantee>
  <Permission xmlns="">READ_ACP</Permission>
</Grant>
</AccessControlList>
</AccessControlPolicy>

```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-acl	개체에 적용할 미리 제공된(canned) ACL. Values : private or public-read or public-read-write or authenticated-read	
x-amz-grant-full-control	허가자에게 개체에 대한 READ, READ_ACP, WRITE_ACP 권한을 부여합니다.	
x-amz-grant-read	부여자가 개체 데이터와 해당 메타데이터를 읽을 수 있습니다.	
x-amz-grant-read-acp	허가자가 개체 ACL을 읽을 수 있습니다.	
x-amz-grant-write	허가자가 버킷에 새 개체를 만들 수 있습니다.	
x-amz-grant-write-acp	허가자가 해당 개체에 대한 ACL을 쓸 수 있습니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(엑세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

```

<?xml version="1.0" encoding="UTF-8"?>
<AccessControlPolicy xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <AccessControlList>
    <Grant>
      <Grantee>
        <DisplayName>string</DisplayName>
        <EmailAddress>string</EmailAddress>
        <ID>string</ID>
        <xsi:type>string</xsi:type>
        <URI>string</URI>
      </Grantee>
      <Permission>string</Permission>
    </Grant>
  </AccessControlList>
</AccessControlPolicy>

```

```
</Grant>
</AccessControlList>
<Owner>
  <DisplayName>string</DisplayName>
  <ID>string</ID>
</Owner>
</AccessControlPolicy>
```

# Response Syntax

HTTP/1.1 200

# Response Elements

# Sample Response

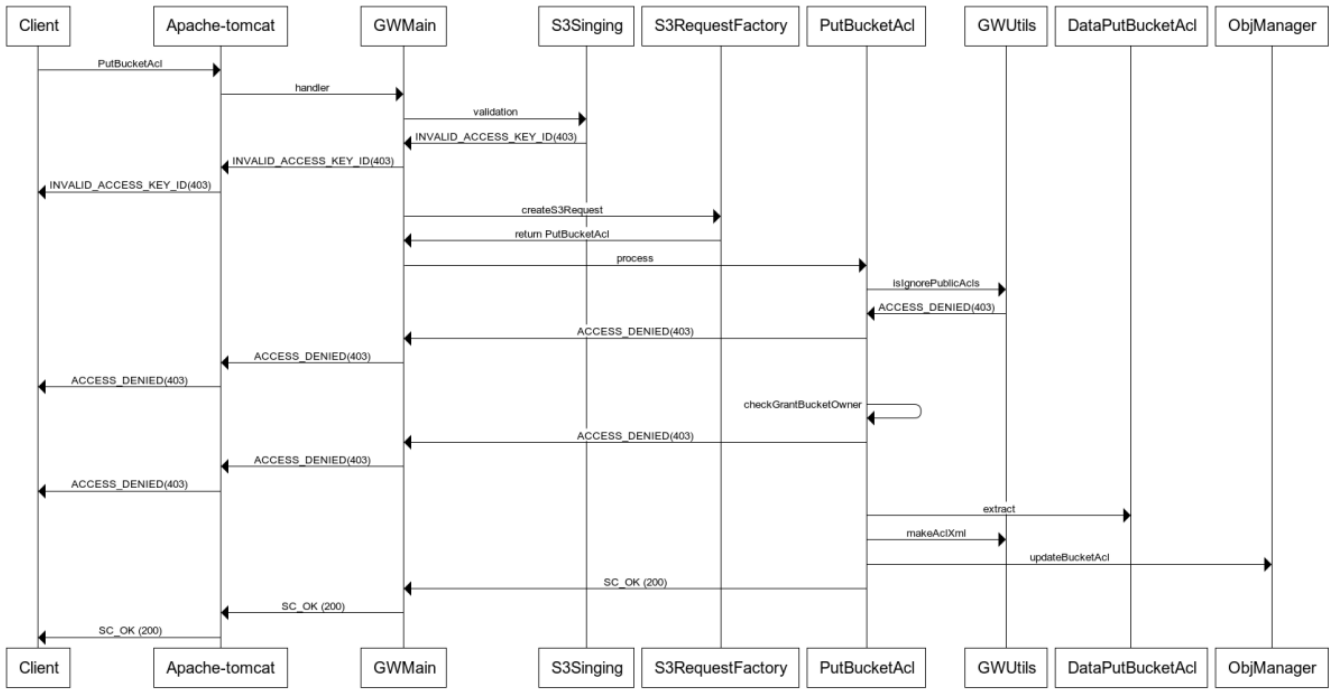
```
HTTP/1.1 200 OK
x-amz-id-2: Nxq03PNiMHXXGwjgv15LLgUoAmPVmG0xtZw2sxePXLhpIvcyouXDrcQUawWXc0K0
x-amz-request-id: C651BC9B4E1BD401
Date: Thu, 12 Apr 2012 20:04:28 GMT
Content-Length: 0
Server: KSAN-GW
```

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests





# PutBucketCors

## Overview

버킷의 CORS(Cross-Origin Resource Sharing) 구성을 설정합니다.

## Request Syntax

```
PUT /Bucket/?cors HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <CORSRule>
    <AllowedHeader>string</AllowedHeader>
    ...
    <AllowedMethod>string</AllowedMethod>
    ...
    <AllowedOrigin>string</AllowedOrigin>
    ...
    <ExposeHeader>string</ExposeHeader>
    ...
    <ID>string</ID>
    <MaxAgeSeconds>integer</MaxAgeSeconds>
  </CORSRule>
  ...
</CORSConfiguration>
```

## Sample Request

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <CORSRule>
    <AllowedHeader>string</AllowedHeader>
    ...
    <AllowedMethod>string</AllowedMethod>
    ...
    <AllowedOrigin>string</AllowedOrigin>
    ...
  </CORSRule>
  ...
</CORSConfiguration>
```

```

    <ExposeHeader>string</ExposeHeader>
    ...
    <ID>string</ID>
    <MaxAgeSeconds>integer</MaxAgeSeconds>
  </CORSRule>
  ...
</CORSConfiguration>

```

# Response Syntax

HTTP/1.1 200

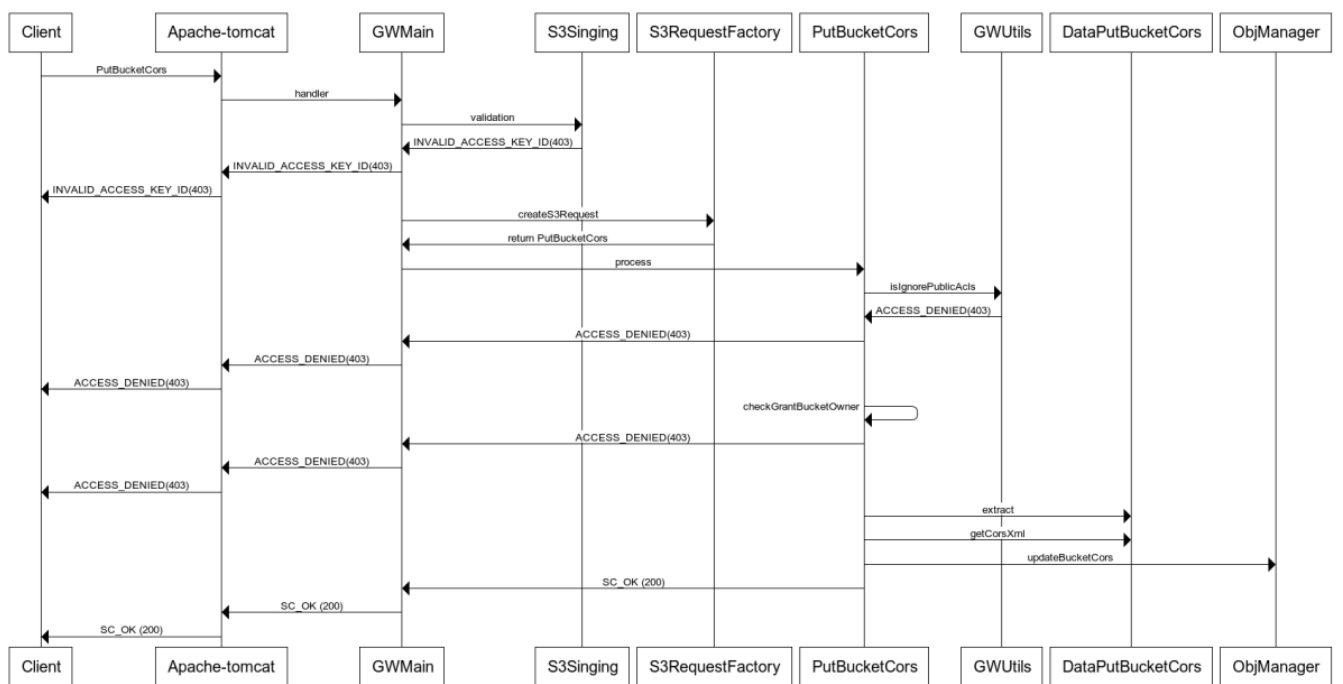
# Response Elements

# Sample Response

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# PutBucketEncryption

## Overview

버킷에 대한 기본 암호화 설정을 구성합니다.

## Request Syntax

```
PUT /Bucket/?encryption HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<ServerSideEncryptionConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ApplyServerSideEncryptionByDefault>
      <KMSEncryptionKeyID>string</KMSEncryptionKeyID>
      <SSEAlgorithm>string</SSEAlgorithm>
    </ApplyServerSideEncryptionByDefault>
    <BucketKeyEnabled>boolean</BucketKeyEnabled>
  </Rule>
  ...
</ServerSideEncryptionConfiguration>
```

## Sample Request

```
PUT /ExampleBucket/?encryption HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 06 Sep 2017 12:00:00 GMT
Authorization: authorization
Content-Length: length

<ServerSideEncryptionConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ApplyServerSideEncryptionByDefault>
      <SSEAlgorithm>aws:kms</SSEAlgorithm>
      <KMSEncryptionKeyID>arn:aws:kms:us-east-1:1234:5678example</KMSEncryptionKeyID>
    </ApplyServerSideEncryptionByDefault>
  </Rule>
</ServerSideEncryptionConfiguration>
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<ServerSideEncryptionConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <ApplyServerSideEncryptionByDefault>
      <KMSEncryptionKeyID>string</KMSEncryptionKeyID>
      <SSEAlgorithm>string</SSEAlgorithm>
    </ApplyServerSideEncryptionByDefault>
    <BucketKeyEnabled>boolean</BucketKeyEnabled>
  </Rule>
  ...
</ServerSideEncryptionConfiguration>
```

# Response Syntax

HTTP/1.1 200

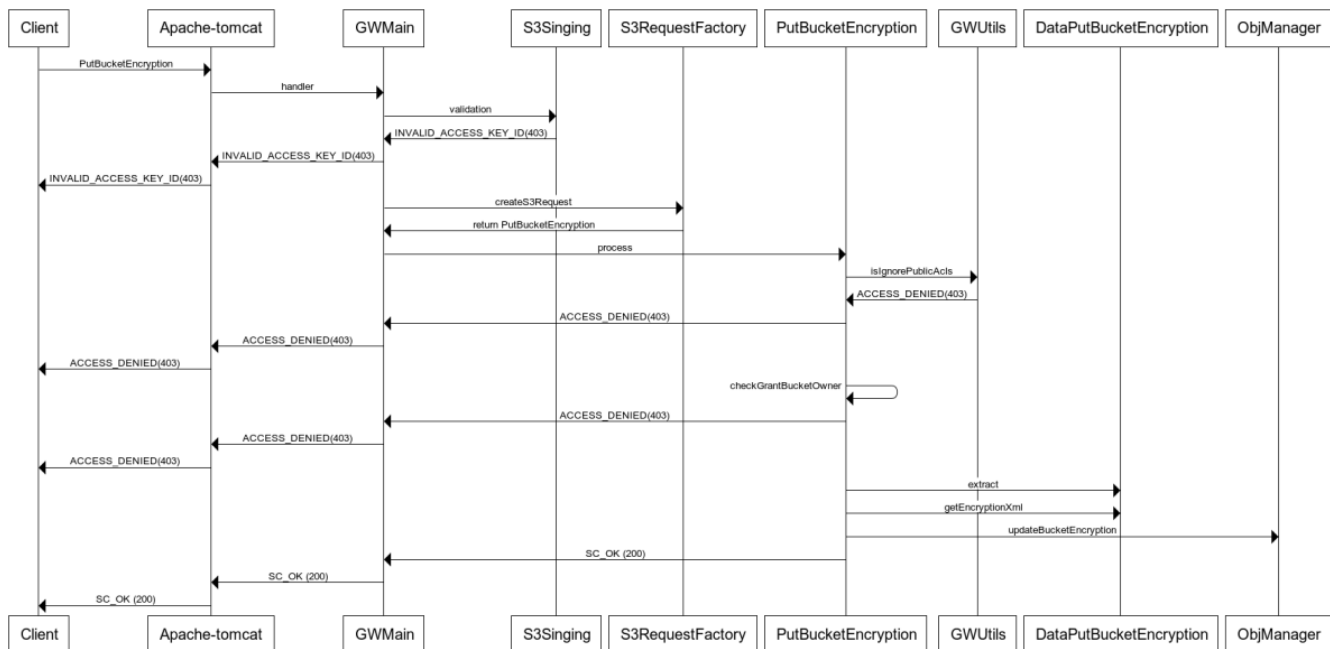
# Response Elements

# Sample Response

# HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# PutBucketLifecycleConfiguration

## Overview

버킷에 대한 새 수명 주기 구성을 만들거나 기존 수명 주기 구성을 바꿉니다.

## Request Syntax

```
PUT /Bucket/?lifecycle HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <AbortIncompleteMultipartUpload>
      <DaysAfterInitiation>integer</DaysAfterInitiation>
    </AbortIncompleteMultipartUpload>
    <Expiration>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <ExpiredObjectDeleteMarker>boolean</ExpiredObjectDeleteMarker>
    </Expiration>
    <Filter>
      <And>
        <ObjectSizeGreaterThan>long</ObjectSizeGreaterThan>
        <ObjectSizeLessThan>long</ObjectSizeLessThan>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
        ...
      </And>
      <ObjectSizeGreaterThan>long</ObjectSizeGreaterThan>
      <ObjectSizeLessThan>long</ObjectSizeLessThan>
      <Prefix>string</Prefix>
      <Tag>
        <Key>string</Key>
        <Value>string</Value>
      </Tag>
    </Filter>
    <ID>string</ID>
    <NoncurrentVersionExpiration>
      <NewerNoncurrentVersions>integer</NewerNoncurrentVersions>
      <NoncurrentDays>integer</NoncurrentDays>
    </NoncurrentVersionExpiration>
    <NoncurrentVersionTransition>
      <NewerNoncurrentVersions>integer</NewerNoncurrentVersions>
      <NoncurrentDays>integer</NoncurrentDays>
      <StorageClass>string</StorageClass>
    </NoncurrentVersionTransition>
    ...
    <Prefix>string</Prefix>
    <Status>string</Status>
    <Transition>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <StorageClass>string</StorageClass>
    </Transition>
    ...
  </Rule>
  ...
</LifecycleConfiguration>
```

# Sample Request

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<LifecycleConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Rule>
    <AbortIncompleteMultipartUpload>
      <DaysAfterInitiation>integer</DaysAfterInitiation>
    </AbortIncompleteMultipartUpload>
    <Expiration>
      <Date>timestamp</Date>
      <Days>integer</Days>
      <ExpiredObjectDeleteMarker>boolean</ExpiredObjectDeleteMarker>
    </Expiration>
    <Filter>
      <And>
        <ObjectSizeGreaterThan>long</ObjectSizeGreaterThan>
        <ObjectSizeLessThan>long</ObjectSizeLessThan>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
        ...
      </And>
      <ObjectSizeGreaterThan>long</ObjectSizeGreaterThan>
      <ObjectSizeLessThan>long</ObjectSizeLessThan>
      <Prefix>string</Prefix>
      <Tag>
        <Key>string</Key>
        <Value>string</Value>
      </Tag>
    </Filter>
    <ID>string</ID>
    <NoncurrentVersionExpiration>
      <NewerNoncurrentVersions>integer</NewerNoncurrentVersions>
      <NoncurrentDays>integer</NoncurrentDays>
    </NoncurrentVersionExpiration>
    <NoncurrentVersionTransition>
      <NewerNoncurrentVersions>integer</NewerNoncurrentVersions>
      <NoncurrentDays>integer</NoncurrentDays>
      <StorageClass>string</StorageClass>
    </NoncurrentVersionTransition>
    ...
    <Prefix>string</Prefix>
    <Status>string</Status>
    <Transition>
```



```

    <Date>timestamp</Date>
    <Days>integer</Days>
    <StorageClass>string</StorageClass>
  </Transition>
  ...
</Rule>
...
</LifecycleConfiguration>

```

## Response Syntax

HTTP/1.1 200

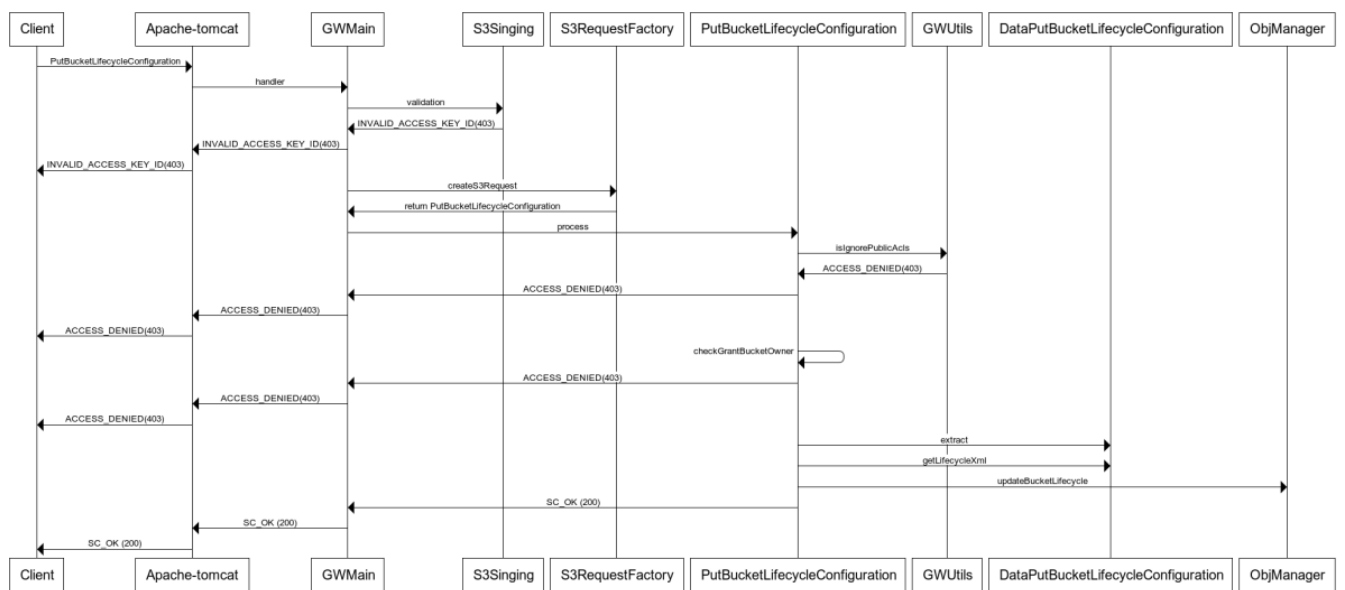
## Response Elements

## Sample Response

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# PutBucketPolicy

## Overview

버킷 정책을 S3 버킷에 적용합니다.

## Request Syntax

```
PUT /Bucket/?policy HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-MD5: ContentMD5
x-amz-confirm-remove-self-bucket-access: ConfirmRemoveSelfBucketAccess
x-amz-expected-bucket-owner: ExpectedBucketOwner

{ Policy in JSON format }
```

## Sample Request

```
PUT /ExampleBucket/?policy HTTP/1.1
Host: 192.168.111.111:8080
Date: Tue, 04 Apr 2010 20:34:56 GMT
Authorization: authorization string

{
  "Version": "2008-10-17",
  "Id": "aaaa-bbbb-cccc-dddd",
  "Statement" : [
    {
      "Effect": "Allow",
      "Sid": "1",
      "Principal" : {
        "AWS": [ "111122223333", "444455556666" ]
      },
      "Action": [ "s3:*" ],
      "Resource": "arn:aws:s3:::bucket/*"
    }
  ]
}
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-confirm-remove-self-bucket-access	이 버킷 정책을 변경할 사용 권한을 제거하려면 이 매개 변수를 true로 설정하십시오.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

Parameter Name	Description
Policy	Response Syntax 참조

# Response Syntax

HTTP/1.1 200

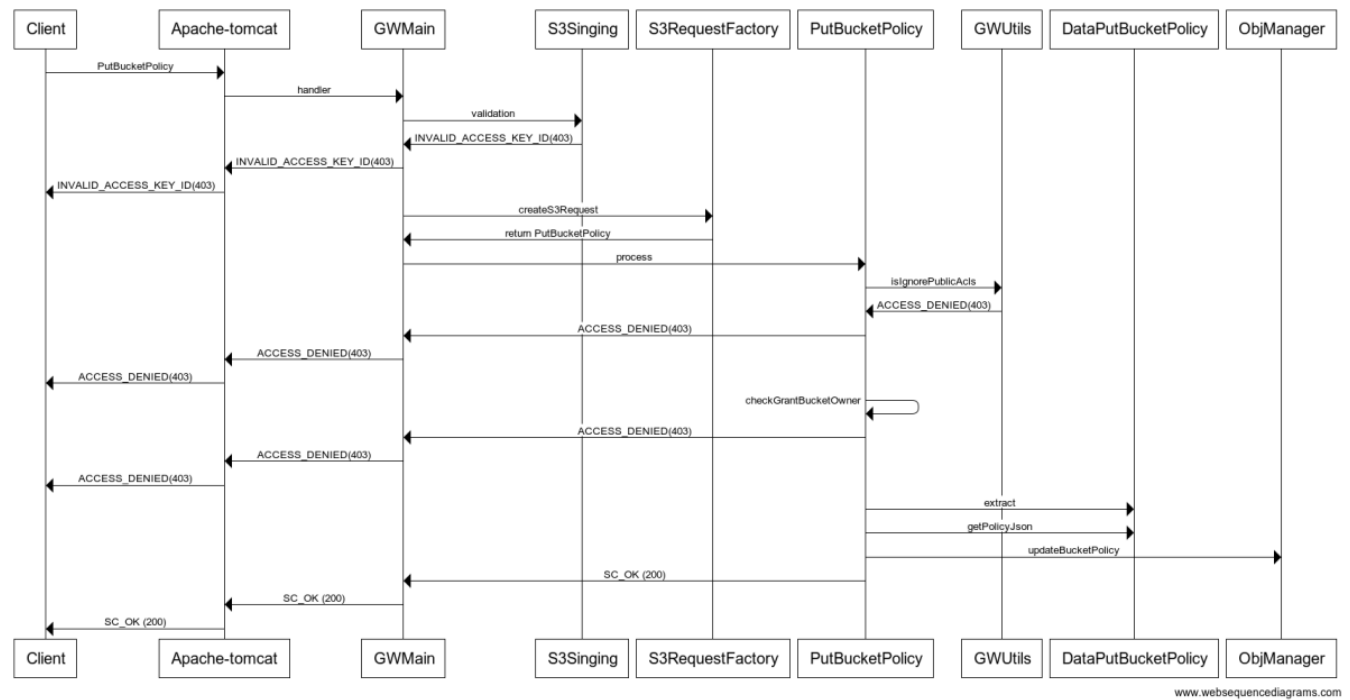
# Response Elements

# Sample Response

# HTTP Status Code

Status Code	Description
204 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# PutBucketReplication

## Overview

복제 구성을 만들거나 기존 복제 구성을 바꿉니다.

## Request Syntax

```
PUT /Bucket/?replication HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-MD5: ContentMD5
x-amz-bucket-object-lock-token: Token
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<ReplicationConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Role>string</Role>
  <Rule>
    <DeleteMarkerReplication>
      <Status>string</Status>
    </DeleteMarkerReplication>
    <Destination>
      <AccessControlTranslation>
        <Owner>string</Owner>
      </AccessControlTranslation>
      <Account>string</Account>
      <Bucket>string</Bucket>
      <EncryptionConfiguration>
        <ReplicaKmsKeyID>string</ReplicaKmsKeyID>
      </EncryptionConfiguration>
      <Metrics>
        <EventThreshold>
          <Minutes>integer</Minutes>
        </EventThreshold>
        <Status>string</Status>
      </Metrics>
      <ReplicationTime>
        <Status>string</Status>
        <Time>
          <Minutes>integer</Minutes>
        </Time>
      </ReplicationTime>
      <StorageClass>string</StorageClass>
    </Destination>
    <ExistingObjectReplication>
      <Status>string</Status>
    </ExistingObjectReplication>
    <Filter>
      <And>
        <Prefix>string</Prefix>
        <Tag>
          <Key>string</Key>
          <Value>string</Value>
        </Tag>
        ...
      </And>
      <Prefix>string</Prefix>
      <Tag>
        <Key>string</Key>
        <Value>string</Value>
      </Tag>
    </Filter>
    <ID>string</ID>
```

```

    <Prefix>string</Prefix>
    <Priority>integer</Priority>
    <SourceSelectionCriteria>
      <ReplicaModifications>
        <Status>string</Status>
      </ReplicaModifications>
      <SseKmsEncryptedObjects>
        <Status>string</Status>
      </SseKmsEncryptedObjects>
    </SourceSelectionCriteria>
    <Status>string</Status>
  </Rule>
  ...
</ReplicationConfiguration>

```

## Sample Request

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-bucket-object-lock-token	기존 버킷에 대해 개체 잠금을 사용할 수 있는 토큰입니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

```

<ReplicationConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Role>string</Role>
  <Rule>
    <DeleteMarkerReplication>
      <Status>string</Status>
    </DeleteMarkerReplication>
    <Destination>
      <AccessControlTranslation>
        <Owner>string</Owner>
      </AccessControlTranslation>
      <Account>string</Account>
      <Bucket>string</Bucket>
      <EncryptionConfiguration>
        <ReplicaKmsKeyID>string</ReplicaKmsKeyID>
      </EncryptionConfiguration>
      <Metrics>
        <EventThreshold>
          <Minutes>integer</Minutes>
        </EventThreshold>
        <Status>string</Status>
      </Metrics>
      <ReplicationTime>
        <Status>string</Status>
        <Time>
          <Minutes>integer</Minutes>
        </Time>
      </ReplicationTime>
    </Destination>
  </Rule>
</ReplicationConfiguration>

```

```
<StorageClass>string</StorageClass>
</Destination>
<ExistingObjectReplication>
  <Status>string</Status>
</ExistingObjectReplication>
<Filter>
  <And>
    <Prefix>string</Prefix>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
    ...
  </And>
  <Prefix>string</Prefix>
  <Tag>
    <Key>string</Key>
    <Value>string</Value>
  </Tag>
</Filter>
<ID>string</ID>
<Prefix>string</Prefix>
<Priority>integer</Priority>
<SourceSelectionCriteria>
  <ReplicaModifications>
    <Status>string</Status>
  </ReplicaModifications>
  <SseKmsEncryptedObjects>
    <Status>string</Status>
  </SseKmsEncryptedObjects>
</SourceSelectionCriteria>
<Status>string</Status>
</Rule>
...
</ReplicationConfiguration>
```

# Response Syntax

HTTP/1.1 200

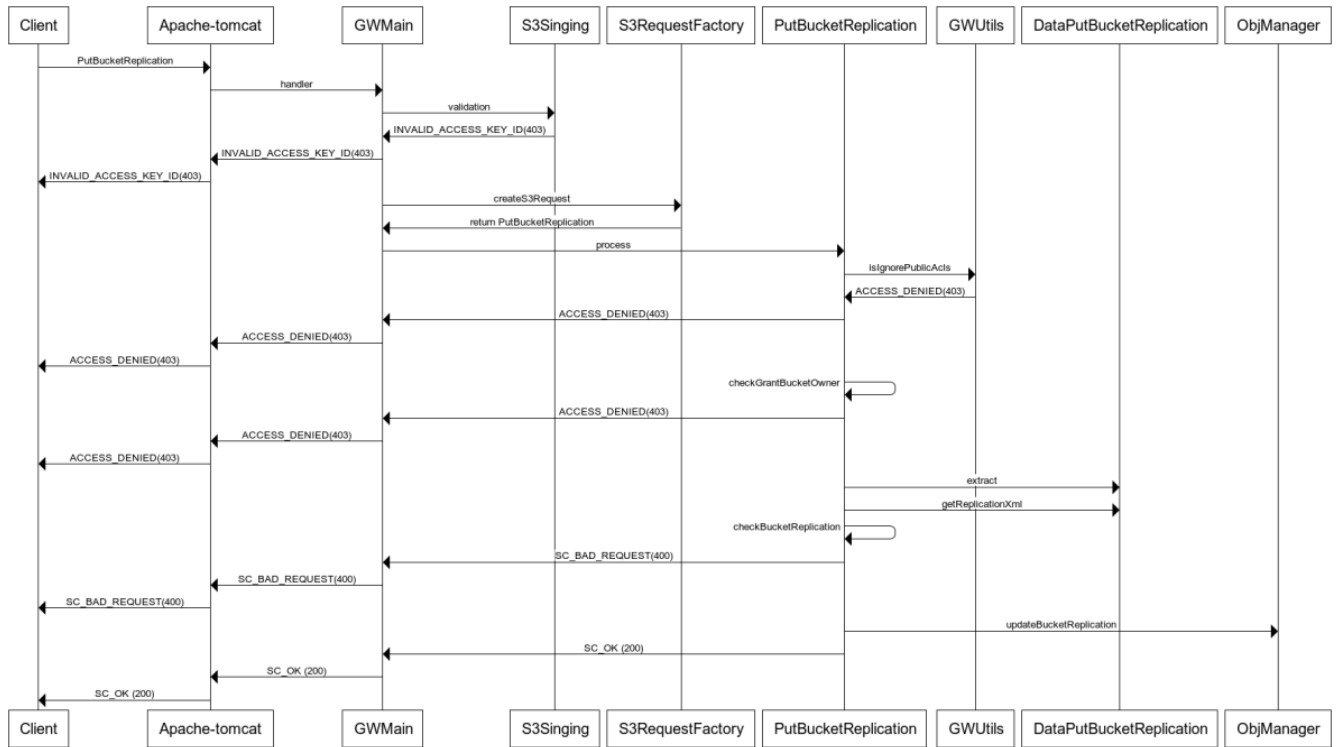
## Response Elements

## Sample Response

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
400 (Bad Request)	요청 형식이 맞지 않음
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# PutBucketTagging

## Overview

버킷의 태그를 설정합니다.

## Request Syntax

```
PUT /Bucket/?tagging HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<Tagging xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>
```

## Sample Request

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<Tagging xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>
```

## Response Syntax



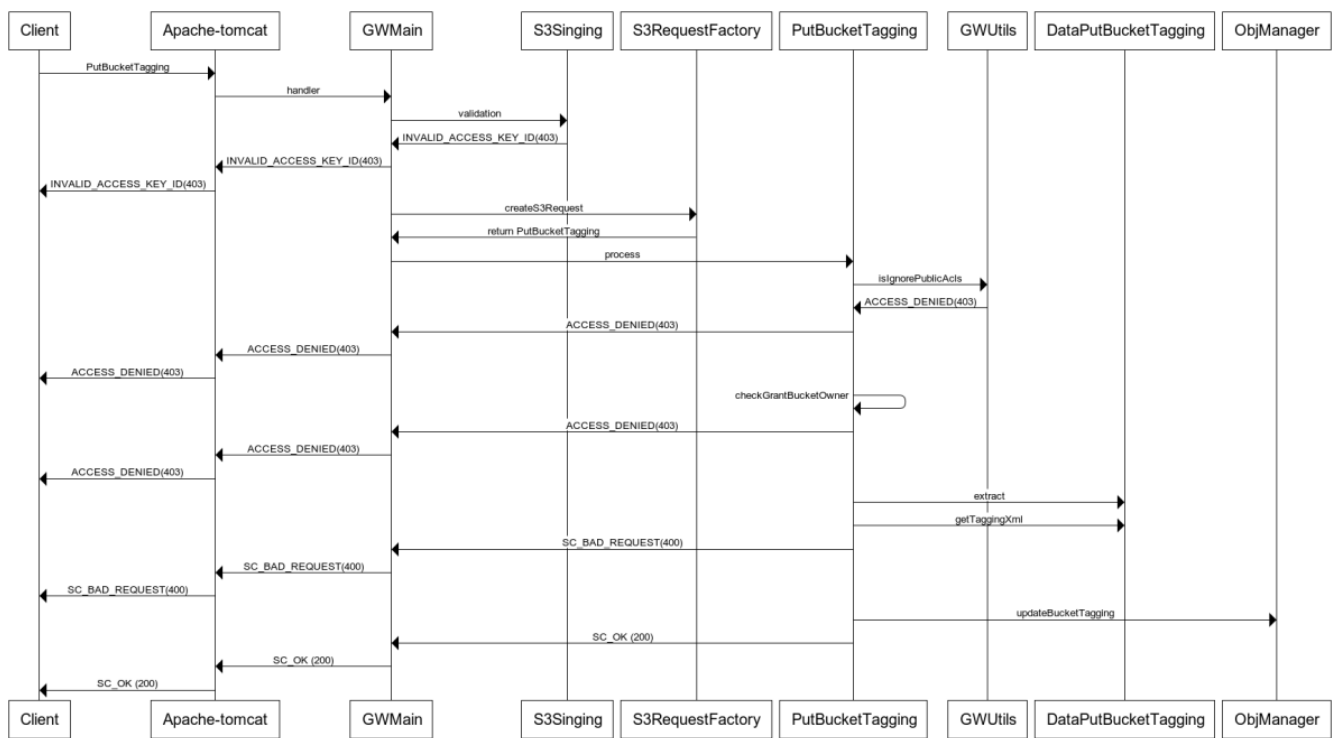
# Response Elements

## Sample Response

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# PutBucketVersioning

## Overview

기존 버킷의 버전 상태를 설정합니다.

## Request Syntax

```
PUT /Bucket/?versioning HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-MD5: ContentMD5
x-amz-mfa: MFA
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<VersioningConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <MfaDelete>string</MfaDelete>
  <Status>string</Status>
</VersioningConfiguration>
```

## Sample Request

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-mfa	인증 장치의 일련 번호, 공백 및 인증 장치에 표시되는 값의 연결.	

## Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<VersioningConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <MfaDelete>string</MfaDelete>
  <Status>string</Status>
</VersioningConfiguration>
```

## Response Syntax

```
HTTP/1.1 200
```

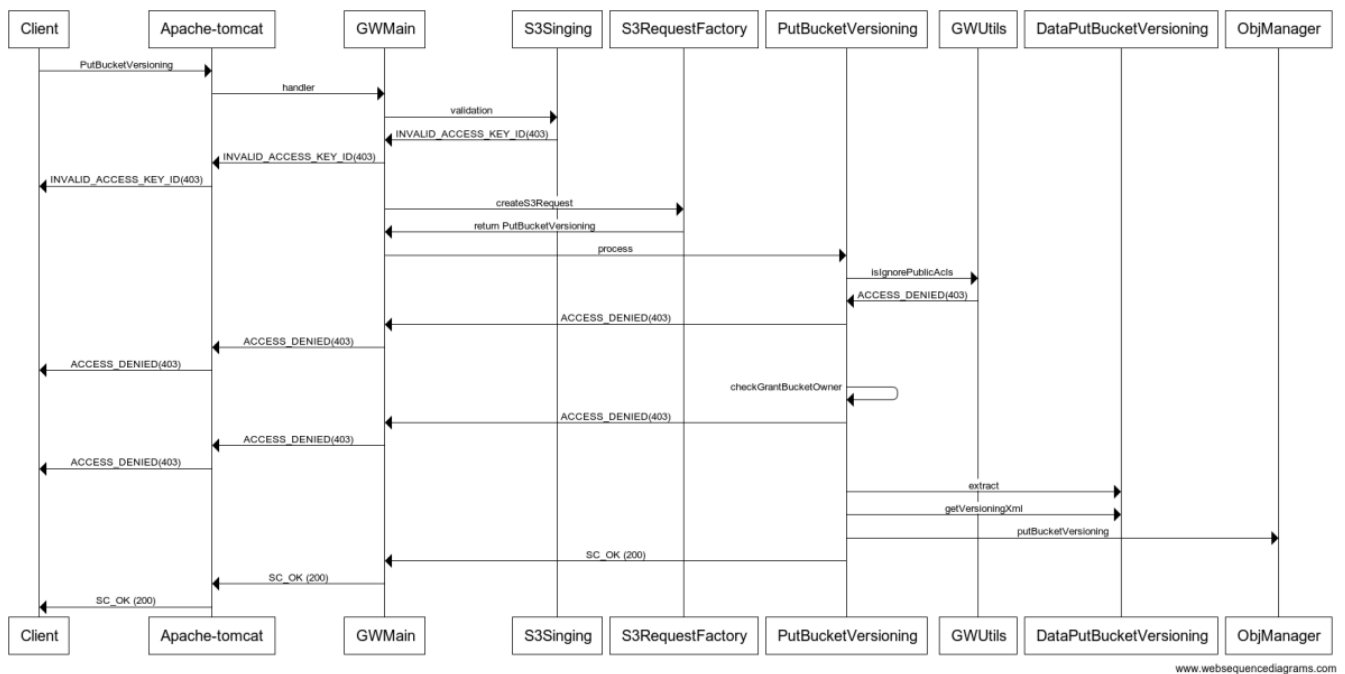
## Response Elements

# Sample Response

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# PutBucketWebsite

## Overview

웹 사이트의 구성을 설정합니다.

## Request Syntax

```
PUT /Bucket/?website HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<WebsiteConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <ErrorDocument>
    <Key>string</Key>
  </ErrorDocument>
  <IndexDocument>
    <Suffix>string</Suffix>
  </IndexDocument>
  <RedirectAllRequestsTo>
    <HostName>string</HostName>
    <Protocol>string</Protocol>
  </RedirectAllRequestsTo>
  <RoutingRules>
    <RoutingRule>
      <Condition>
        <HttpErrorCodeReturnedEquals>string</HttpErrorCodeReturnedEquals>
        <KeyPrefixEquals>string</KeyPrefixEquals>
      </Condition>
      <Redirect>
        <HostName>string</HostName>
        <HttpRedirectCode>string</HttpRedirectCode>
        <Protocol>string</Protocol>
        <ReplaceKeyPrefixWith>string</ReplaceKeyPrefixWith>
        <ReplaceKeyWith>string</ReplaceKeyWith>
      </Redirect>
    </RoutingRule>
  </RoutingRules>
</WebsiteConfiguration>
```

## Sample Request

```
PUT /ExampleBucket/?website HTTP/1.1
Host: 192.168.111.111:8080
Content-Length: 256
Date: Thu, 27 Jan 2011 12:00:00 GMT
Authorization: signatureValue

<WebsiteConfiguration xmlns='http://s3.amazonaws.com/doc/2006-03-01/'>
  <IndexDocument>
    <Suffix>index.html</Suffix>
  </IndexDocument>
  <ErrorDocument>
    <Key>SomeErrorDocument.html</Key>
  </ErrorDocument>
</WebsiteConfiguration>
```

# URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

# Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<WebsiteConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <ErrorDocument>
    <Key>string</Key>
  </ErrorDocument>
  <IndexDocument>
    <Suffix>string</Suffix>
  </IndexDocument>
  <RedirectAllRequestsTo>
    <HostName>string</HostName>
    <Protocol>string</Protocol>
  </RedirectAllRequestsTo>
  <RoutingRules>
    <RoutingRule>
      <Condition>
        <HttpErrorCodeReturnedEquals>string</HttpErrorCodeReturnedEquals>
        <KeyPrefixEquals>string</KeyPrefixEquals>
      </Condition>
      <Redirect>
        <HostName>string</HostName>
        <HttpRedirectCode>string</HttpRedirectCode>
        <Protocol>string</Protocol>
        <ReplaceKeyPrefixWith>string</ReplaceKeyPrefixWith>
        <ReplaceKeyWith>string</ReplaceKeyWith>
      </Redirect>
    </RoutingRule>
  </RoutingRules>
</WebsiteConfiguration>
```

# Response Syntax

```
HTTP/1.1 200
```

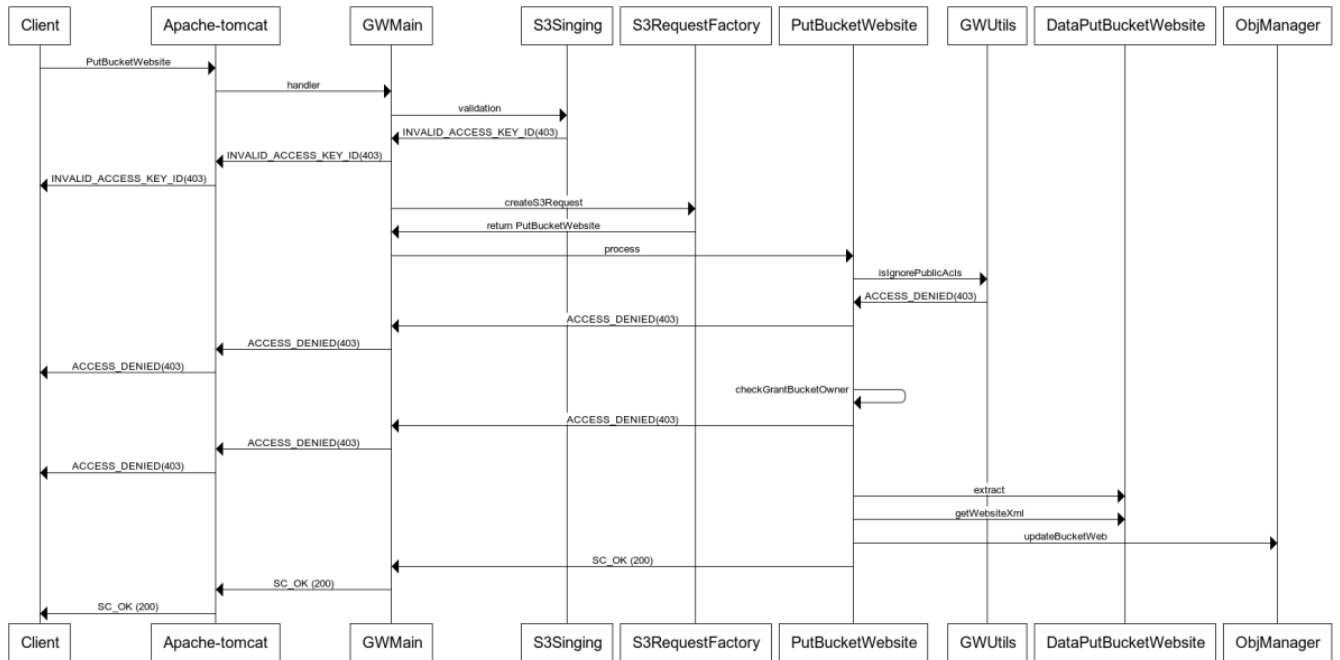
# Response Elements

# Sample Response

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# PutObject

## Overview

버킷에 개체를 추가합니다.

## Request Syntax

```
PUT /Buckst/Key HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-acl: ACL
Cache-Control: CacheControl
Content-Disposition: ContentDisposition
Content-Encoding: ContentEncoding
Content-Language: ContentLanguage
Content-Length: ContentLength
Content-MD5: ContentMD5
Content-Type: ContentType
Expires: Expires
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write-acp: GrantWriteACP
x-amz-server-side-encryption: ServerSideEncryption
x-amz-storage-class: StorageClass
x-amz-website-redirect-location: WebsiteRedirectLocation
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-context: SSEKMSEncryptionContext
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-request-payer: RequestPayer
x-amz-tagging: Tagging
x-amz-object-lock-mode: ObjectLockMode
x-amz-object-lock-retain-until-date: ObjectLockRetainUntilDate
x-amz-object-lock-legal-hold: ObjectLockLegalHoldStatus
x-amz-expected-bucket-owner: ExpectedBucketOwner

Body
```

## Sample Request

```
PUT /ExampleBucket/my-image.jpg HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: authorization string
Content-Type: text/plain
Content-Length: 11434
x-amz-meta-author: Janet
Expect: 100-continue
[11434 bytes of object data]
```

## URI request Parameters

Parameter Name	Description	Required
----------------	-------------	----------

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	PUT 작업이 시작된 개체 키입니다.	Yes
Cache-Control	요청/응답 체인을 따라 캐싱 동작을 지정합니다.	
Content-Disposition	개체에 대한 프레젠테이션 정보를 지정합니다.	
Content-Encoding	개체에 적용된 콘텐츠 인코딩과 콘텐츠 유형 헤더 필드가 참조하는 미디어 유형을 가져오기 위해 적용해야 하는 디코딩 메커니즘을 지정합니다.	
Content-Language	콘텐츠가 사용되는 언어입니다.	
Content-Type	내용 유형	
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
Content-Length	본문의 크기(바이트)입니다.	
Expires	개체를 더 이상 캐시할 수 없는 날짜 및 시간입니다.	
x-amz-acl	개체에 적용할 미리 제공된(canned) ACL.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-grant-full-control	허가자에게 개체에 대한 READ, READ_ACL, WRITE_ACP 권한을 부여합니다.	
x-amz-grant-read	부여자가 개체 데이터와 해당 메타데이터를 읽을 수 있습니다.	
x-amz-grant-read-acp	허가자가 개체 ACL을 읽을 수 있습니다.	
x-amz-grant-write	허가자가 버킷에 새 개체를 만들 수 있습니다.	
x-amz-grant-write-acp	허가자가 해당 개체에 대한 ACL을 쓸 수 있습니다.	
x-amz-metadata-directive	메타데이터를 원본 개체에서 복사할지 아니면 요청에 제공된 메타데이터로 바꿀지를 지정합니다. Values : COPY or REPLACE	
x-amz-object-lock-legal-hold	복사된 개체에 증거 보존을 적용할지 여부를 지정합니다. Values : ON or OFF	
x-amz-object-lock-mode	복사된 개체에 적용할 개체 Lock 모드입니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	
x-amz-server-side-encryption	S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘(예: AES256, aws:kms) Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	개체 암호화에 사용할 AWS KMS 키 ID를 지정합니다. SSL을 통하거나 SigV4를 사용하지 않으면 AWS KMS로 보호되는 개체에 대한 모든 GET 및 PUT 요청이 실패합니다. 	
x-amz-server-side-encryption-bucket-key-enabled	AWS KMS(SSE-KMS)를 사용한 서버측 암호화의 개체 암호화에 Amazon S3 버킷 키를 사용해야 하는지 여부를 지정합니다.	
x-amz-server-side-encryption-context	개체 암호화에 사용할 AWS KMS 암호화 컨텍스트를 지정합니다.	
x-amz-server-side-encryption-customer-algorithm	개체를 암호화할 때 사용할 알고리즘(예: AES256)을 지정합니다.	
x-amz-server-side-encryption-customer-key	데이터 암호화에 사용할 S3에 대해 고객이 제공한 암호화 키를 지정합니다. 	
x-amz-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 128비트 MD5 다이제스트를 지정합니다. 	
x-amz-source-expected-bucket-owner	예상 소스 버킷 소유자의 계정 ID입니다.	
x-amz-storage-class	기본적으로 S3는 STANDARD 스토리지 클래스를 사용하여 새로 생성된 개체를 저장합니다.	
x-amz-tagging	개체 대상 개체의 태그 집합은 TaggingDirective와 함께 사용해야 합니다. 태그 집합은 URL 쿼리 매개 변수로 인코딩되어야 합니다.	
x-amz-website-redirect-location	버킷이 웹 사이트로 구성된 경우 이 개체에 대한 요청을 동일한 버킷의 다른 개체 또는 외부 URL로 리디렉션합니다.	



# Request Body

- binary data

## Response Syntax

```
HTTP/1.1 200
x-amz-expiration: Expiration
ETag: ETag
x-amz-server-side-encryption: ServerSideEncryption
x-amz-version-id: VersionId
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-context: SSEKMSEncryptionContext
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-request-charged: RequestCharged
```

## Response Elements

Parameter Name	Description	Required
ETag	ETAG는 웹 서버가 URL에서 찾은 리소스의 특정 버전에 할당한 식별자입니다.	
x-amz-expiration	개체 만료가 구성된 경우 응답에 이 헤더가 포함됩니다.	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
x-amz-server-side-encryption	Amazon S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘 Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	있는 경우 개체에 사용된 AWS KMS(AWS Key Management Service) 대칭 고객 관리 키의 ID를 지정합니다.	
x-amz-server-side-encryption-bucket-key-enabled	복사된 개체가 AWS KMS(SSE-KMS)와의 서버측 암호화에 S3 버킷 키를 사용하는지 여부를 나타냅니다.	
x-amz-server-side-encryption-context	존재하는 경우 개체 암호화에 사용할 AWS KMS 암호화 컨텍스트를 지정합니다.	
x-amz-server-side-encryption-customer-algorithm	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 사용된 암호화 알고리즘을 확인하는 헤더가 포함됩니다.	
x-amz-server-side-encryption-customer-key-MD5	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 이 헤더가 포함되어 고객이 제공한 암호화 키에 대한 왕복 메시지 무결성 확인을 제공합니다.	
x-amz-version-id	개체의 버전 ID입니다.	

## Sample Response

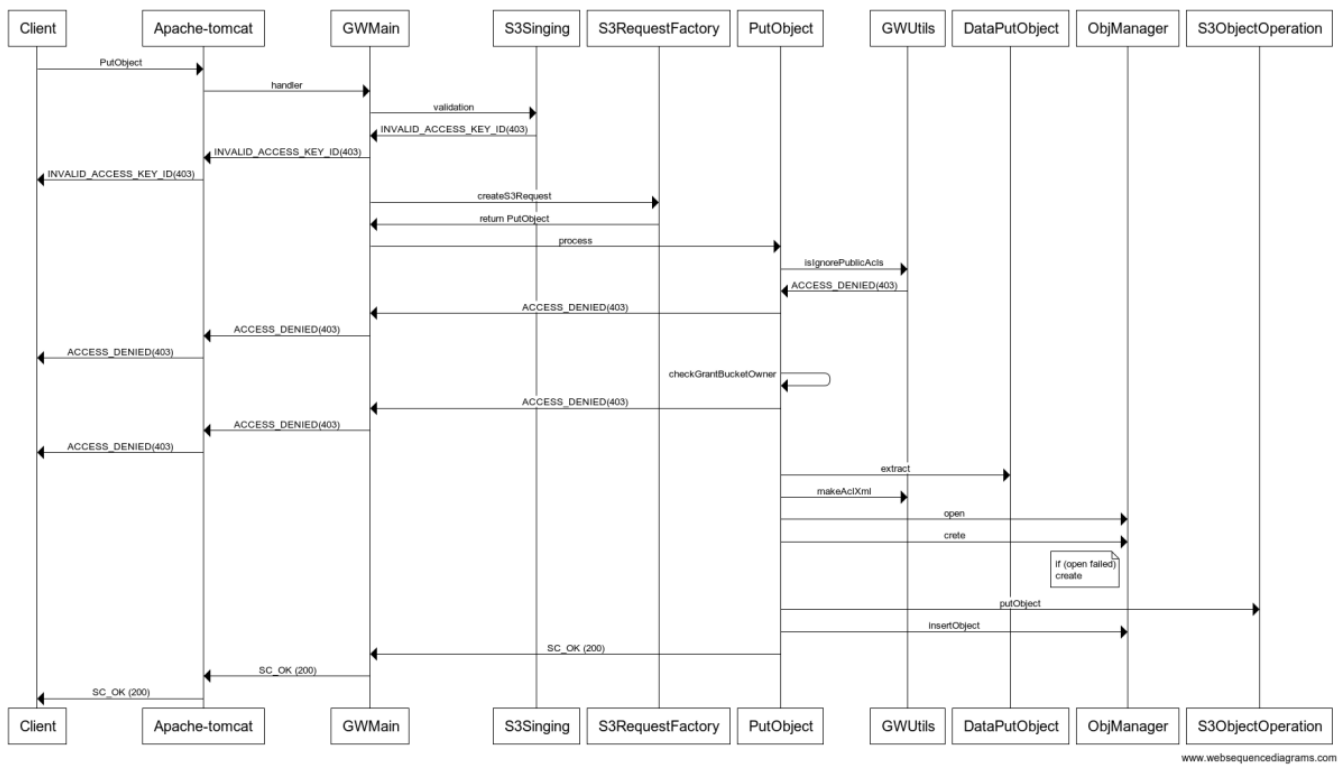
```
HTTP/1.1 100 Continue

HTTP/1.1 200 OK
x-amz-id-2: LriYPLdmOdAiIfgSm/F1YsViT1LW94/xUQxMsF7xiEb1a0wiI0Ix1+zbwZ163pt7
x-amz-request-id: 0A49CE4060975EAC
Date: Wed, 12 Oct 2009 17:50:00 GMT
ETag: "1b2cf535f27731c974343645a3985328"
Content-Length: 0
Connection: close
Server: KSAN-GW
```

# HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

# Processing of requests



# PutObjectAcl

## Overview

S3 버킷의 개체에 대한 ACL(액세스 제어 목록) 권한을 설정합니다.

## Request Syntax

```
PUT /Bucket/Key?acl&versionId=VersionId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-acl: ACL
Content-MD5: ContentMD5
x-amz-grant-full-control: GrantFullControl
x-amz-grant-read: GrantRead
x-amz-grant-read-acp: GrantReadACP
x-amz-grant-write: GrantWrite
x-amz-grant-write-acp: GrantWriteACP
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<AccessControlPolicy xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <AccessControlList>
    <Grant>
      <Grantee>
        <DisplayName>string</DisplayName>
        <EmailAddress>string</EmailAddress>
        <ID>string</ID>
        <xsi:type>string</xsi:type>
        <URI>string</URI>
      </Grantee>
      <Permission>string</Permission>
    </Grant>
  </AccessControlList>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
</AccessControlPolicy>
```

## Sample Request

```
PUT /ExampleBucket/my-image.jpg?acl HTTP/1.1
Host: 192.168.111.111:8080
Date: Wed, 28 Oct 2009 22:32:00 GMT
Authorization: authorization string
Content-Length: 124

<AccessControlPolicy>
  <Owner>
    <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeebf76c078efc7c6caea54ba06a</ID>
    <DisplayName>CustomersName@amazon.com</DisplayName>
  </Owner>
  <AccessControlList>
    <Grant>
      <Grantee xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CanonicalUser">
        <ID>75aa57f09aa0c8caeab4f8c24e99d10f8e7faeeExampleCanonicalUserID</ID>
        <DisplayName>CustomerName@amazon.com</DisplayName>
      </Grantee>
      <Permission>FULL_CONTROL</Permission>
```

```
</Grant>
</AccessControlList>
</AccessControlPolicy>
```

# URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	PUT 작업이 시작된 개체 키입니다.	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
versionId	개체의 특정 버전을 참조하는 데 사용되는 VersionId입니다.	
x-amz-acl	개체에 적용할 미리 제공된(canned) ACL.	
x-amz-source-expected-bucket-owner	예상 소스 버킷 소유자의 계정 ID입니다.	
x-amz-grant-full-control	허가자에게 개체에 대한 READ, READ_ACL, WRITE_ACP 권한을 부여합니다.	
x-amz-grant-read	부여자가 개체 데이터와 해당 메타데이터를 읽을 수 있습니다.	
x-amz-grant-read-acp	허가자가 개체 ACL을 읽을 수 있습니다.	
x-amz-grant-write	허가자가 버킷에 새 개체를 만들 수 있습니다.	
x-amz-grant-write-acp	허가자가 해당 개체에 대한 ACL을 쓸 수 있습니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

# Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<AccessControlPolicy xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <AccessControlList>
    <Grant>
      <Grantee>
        <DisplayName>string</DisplayName>
        <EmailAddress>string</EmailAddress>
        <ID>string</ID>
        <xsi:type>string</xsi:type>
        <URI>string</URI>
      </Grantee>
      <Permission>string</Permission>
    </Grant>
  </AccessControlList>
  <Owner>
    <DisplayName>string</DisplayName>
    <ID>string</ID>
  </Owner>
</AccessControlPolicy>
```

# Response Syntax

```
HTTP/1.1 200
x-amz-request-charged: RequestCharged
```

# Response Elements

Parameter Name	Description	Required
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	

## Sample Response

```

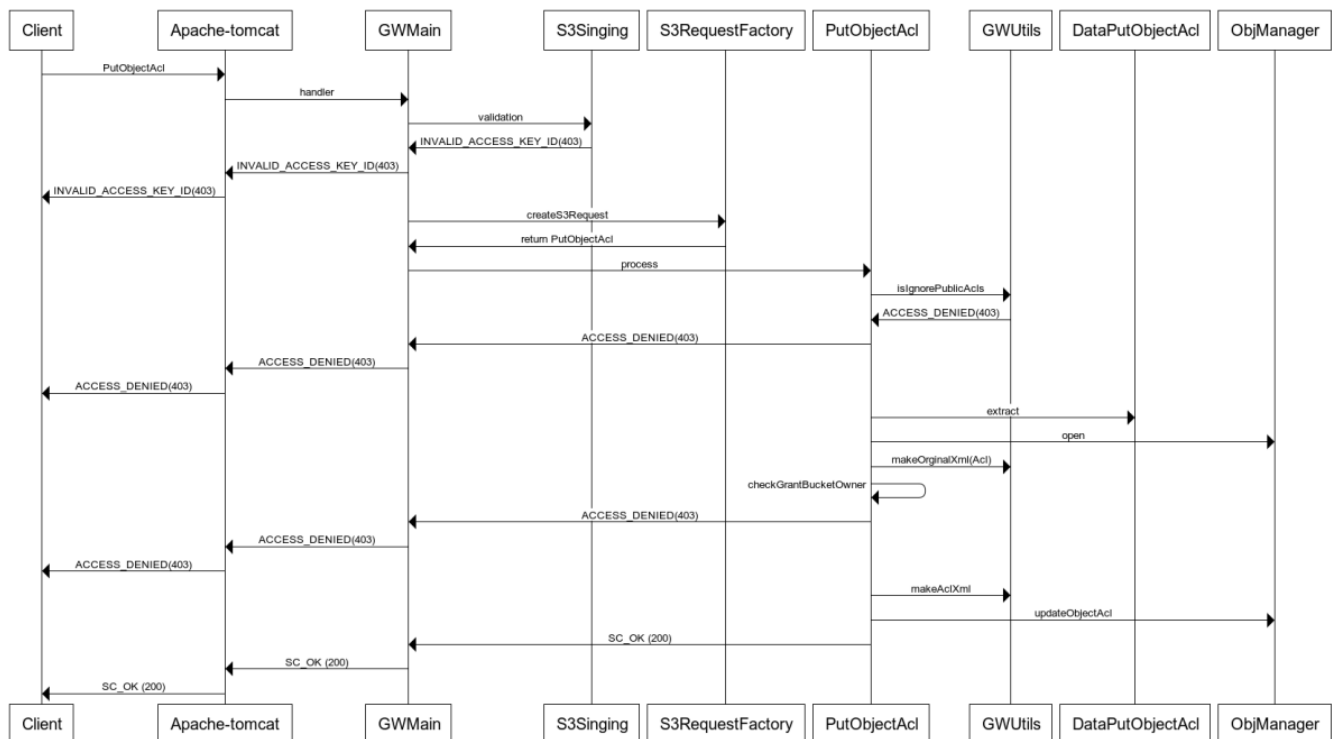
HTTP/1.1 200 OK
x-amz-id-2: eftixk72aD6Ap51T9AS1ed40pIszj7UDNEHGran
x-amz-request-id: 318BC8BC148832E5
x-amz-version-id: 3/L4kqtJlcpXrof3vjVBH40Nr8X8gdRQBpUMLUo
Date: Wed, 28 Oct 2009 22:32:00 GMT
Last-Modified: Sun, 1 Jan 2006 12:00:00 GMT
Content-Length: 0
Connection: close
Server: KSAN-GW

```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# PutObjectLockConfiguration

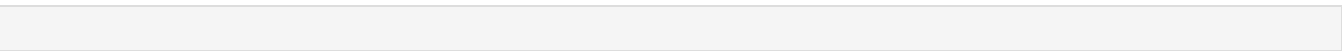
## Overview

지정한 버킷에 개체 잠금 구성을 배치합니다.

## Request Syntax

```
PUT /Bucket/?object-lock HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-request-payer: RequestPayer
x-amz-bucket-object-lock-token: Token
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<ObjectLockConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <ObjectLockEnabled>string</ObjectLockEnabled>
  <Rule>
    <DefaultRetention>
      <Days>integer</Days>
      <Mode>string</Mode>
      <Years>integer</Years>
    </DefaultRetention>
  </Rule>
</ObjectLockConfiguration>
```

## Sample Request



## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-bucket-object-lock-token	기존 버킷에 대해 개체 잠금을 사용할 수 있는 토큰입니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

## Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<ObjectLockConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <ObjectLockEnabled>string</ObjectLockEnabled>
  <Rule>
    <DefaultRetention>
      <Days>integer</Days>
      <Mode>string</Mode>
```

```

    <Years>integer</Years>
  </DefaultRetention>
</Rule>
</ObjectLockConfiguration>

```

## Response Syntax

```

HTTP/1.1 200
x-amz-request-charged: RequestCharged

```

## Response Elements

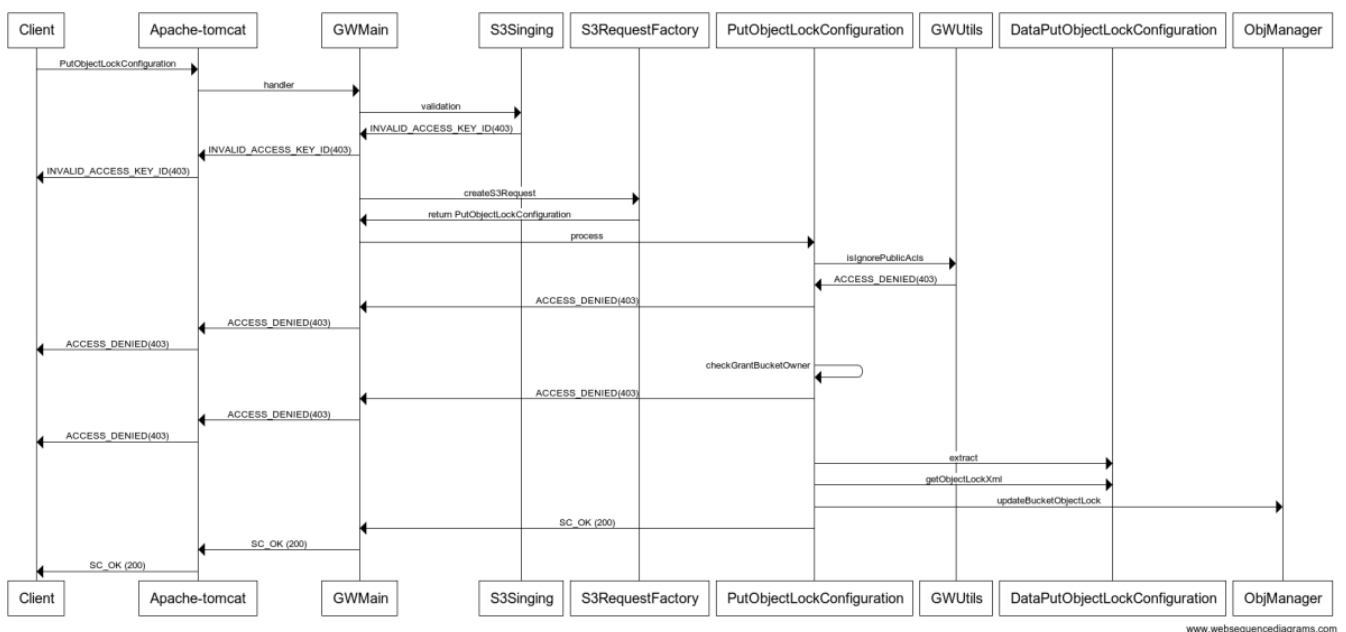
Parameter Name	Description	Required
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	

## Sample Response

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# PutObjectTagging

## Overview

제공된 태그 집합을 개체에 적용합니다.

## Request Syntax

```
PUT /Bucket/Key?tagging&versionId=VersionId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
x-amz-request-payer: RequestPayer
<?xml version="1.0" encoding="UTF-8"?>
<Tagging xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>
```


## Sample Request

```
PUT /ExampleBucket/object-key?tagging HTTP/1.1
Host: 192.168.111.111:8080
Content-Length: length
Content-MD5: pUNXr/BjKK5G2UKExample==
x-amz-date: 20160923T001956Z
Authorization: authorization string
<Tagging>
  <TagSet>
    <Tag>
      <Key>tag1</Key>
      <Value>val1</Value>
    </Tag>
    <Tag>
      <Key>tag2</Key>
      <Value>val2</Value>
    </Tag>
  </TagSet>
</Tagging>
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	PUT 작업이 시작된 개체 키입니다.	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
versionId	개체의 특정 버전을 참조하는 데 사용되는 VersionId입니다.	



Parameter Name	Description	Required
x-amz-bypass-governance-retention	이 작업을 통해 거버넌스 모드 제한을 무시할지 여부를 나타냅니다. 	
x-amz-source-expected-bucket-owner	예상 소스 버킷 소유자의 계정 ID입니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	

## Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<Tagging xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <TagSet>
    <Tag>
      <Key>string</Key>
      <Value>string</Value>
    </Tag>
  </TagSet>
</Tagging>
```

## Response Syntax

```
HTTP/1.1 200
x-amz-version-id: VersionId
```

## Response Elements

Parameter Name	Description	Required
x-amz-version-id	태그 집합이 추가된 개체의 versionId입니다	

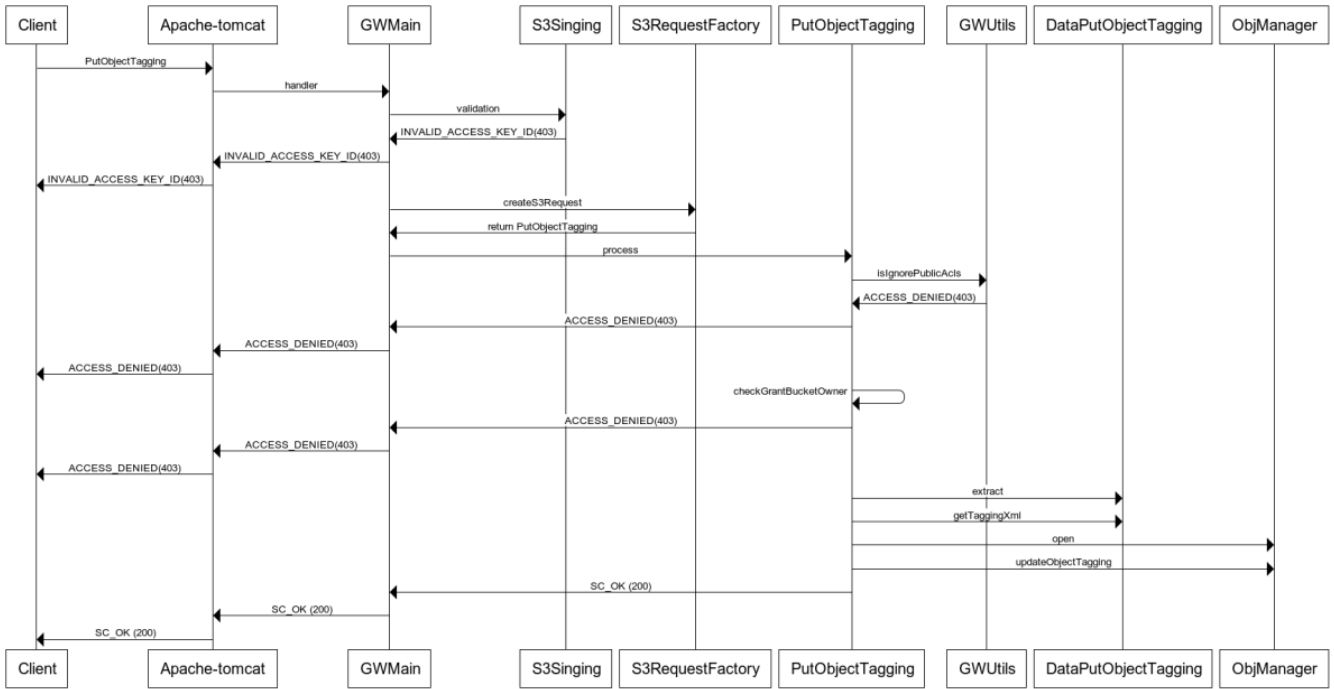
## Sample Response

```
HTTP/1.1 200 OK
x-amz-id-2: YgIPIfBiKa2bj0KMgUAdQkf3ShJT00pXUueF6QKo
x-amz-request-id: 236A8905248E5A01
Date: Fri, 23 Sep 2016 00:20:19 GMT
```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# PutPublicAccessBlock

## Overview

S3 버킷에 대한 PublicAccessBlock 구성을 생성하거나 수정합니다.

## Request Syntax

```
PUT /Bucket/?publicAccessBlock HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-MD5: ContentMD5
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<PublicAccessBlockConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <BlockPublicAcls>boolean</BlockPublicAcls>
  <IgnorePublicAcls>boolean</IgnorePublicAcls>
  <BlockPublicPolicy>boolean</BlockPublicPolicy>
  <RestrictPublicBuckets>boolean</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>
```

## Sample Request

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
x-amz-expected-bucket-owner	대상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	

## Request Body

```
<?xml version="1.0" encoding="UTF-8"?>
<PublicAccessBlockConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <BlockPublicAcls>boolean</BlockPublicAcls>
  <IgnorePublicAcls>boolean</IgnorePublicAcls>
  <BlockPublicPolicy>boolean</BlockPublicPolicy>
  <RestrictPublicBuckets>boolean</RestrictPublicBuckets>
</PublicAccessBlockConfiguration>
```

## Response Syntax

```
HTTP/1.1 200
```

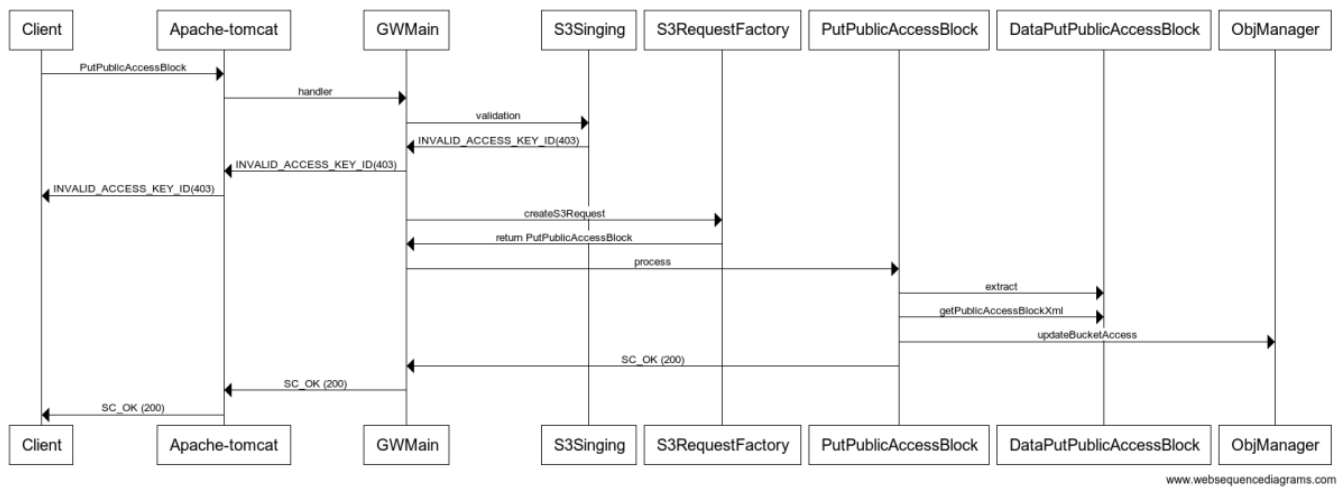
# Response Elements

## Sample Response

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
500 (Server Error)	

## Processing of requests



# UploadPart

## Overview

멀티파트 업로드의 Part를 업로드합니다.

## Request Syntax

```
PUT /Bucket//Key?partNumber=PartNumber&uploadId=UploadId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
Content-Length: ContentLength
Content-MD5: ContentMD5
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
```

Body

## Sample Request

```
PUT /ExampleBucket/my-movie.m2ts?partNumber=1&uploadId=VCVsb2FkIElEIGZvciBlbZZpbmcncyBteS1tb3ZpZS5tMnRzIHVwbG9hZR
HTTP/1.1
Host: 192.168.111.111:8080
Date: Mon, 1 Nov 2010 20:34:56 GMT
Content-Length: 10485760
Content-MD5: pUNXr/BjKK5G2UKvaRRr0A==
Authorization: authorization string

***part data omitted***
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	Multipart Upload가 시작된 개체 키입니다.	Yes
partNumber	업로드 중인 partNumber입니다. 이것은 1에서 10,000 사이의 양의 정수입니다.	
uploadId	Multipart Upload를 식별하는 업로드 ID입니다.	
Content-MD5	데이터의 base64 인코딩 128비트 MD5 다이제스트입니다. 이 헤더를 메시지 무결성 검사로 사용하여 요청 본문이 전송 중에 손상되지 않았는지 확인해야 합니다.	
Content-Length	본문의 크기(바이트)입니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	
x-amz-server-side-encryption-customer-algorithm	개체를 암호화할 때 사용할 알고리즘(예: AES256)을 지정합니다.	
x-amz-server-side-encryption-customer-key	데이터 암호화에 사용할 S3에 대해 고객이 제공한 암호화 키를 지정합니다.	

Parameter Name	Description	Required
x-amz-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 128비트 MD5 다이제스트를 지정합니다.	

## Request Body

- binary data

## Response Syntax

```
HTTP/1.1 200
x-amz-server-side-encryption: ServerSideEncryption
ETag: ETag
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-request-charged: RequestCharged
```

## Response Elements

Parameter Name	Description	Required
ETag	ETAG는 웹 서버가 URL에서 찾은 리소스의 특정 버전에 할당한 식별자입니다.	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
x-amz-server-side-encryption	Amazon S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘 Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	있는 경우 개체에 사용된 AWS KMS(AWS Key Management Service) 대칭 고객 관리 키의 ID를 지정합니다.	
x-amz-server-side-encryption-bucket-key-enabled	복사된 개체가 AWS KMS(SSE-KMS)와의 서버측 암호화에 S3 버킷 키를 사용하는지 여부를 나타냅니다.	
x-amz-server-side-encryption-customer-algorithm	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 사용된 암호화 알고리즘을 확인하는 헤더가 포함됩니다.	
x-amz-server-side-encryption-customer-key-MD5	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 이 헤더가 포함되어 고객이 제공한 암호화 키에 대한 왕복 메시지 무결성 확인을 제공합니다.	

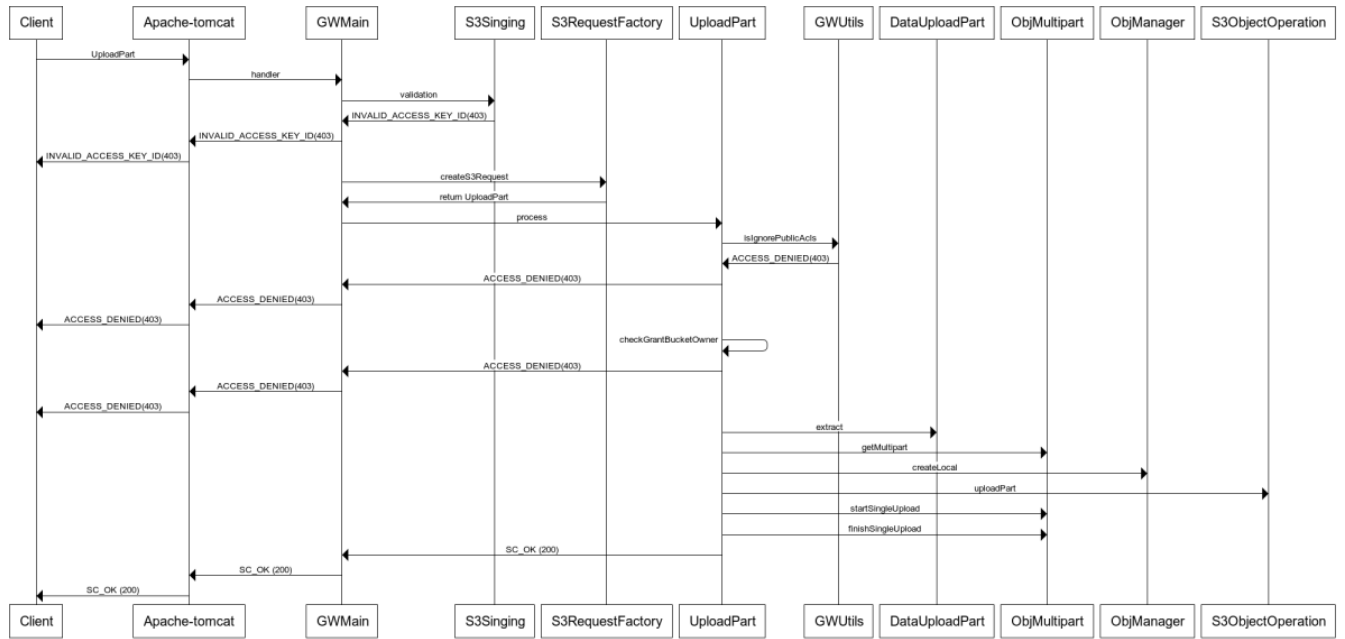
## Sample Response

```
HTTP/1.1 200 OK
x-amz-id-2: Vvag1LuByRx9e6j50nimru9p04ZVKnJ2Qz7/C1NPcfTWAtRPfTa0Fg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 1 Nov 2010 20:34:56 GMT
ETag: "b54357faf0632cce46e942fa68356b38"
Content-Length: 0
Connection: keep-alive
Server: KSAN-GW
```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Such uploadId)	uploadId가 존재하지 않음
500 (Server Error)	

# Processing of requests



# UploadPartCopy

## Overview

기존 객체의 데이터를 복사하여 Part를 업로드합니다.

## Request Syntax

```
PUT /Bucket/Key?partNumber=PartNumber&uploadId=UploadId HTTP/1.1
Host: <KSAN_GW_IP:PORT>
x-amz-copy-source: CopySource
x-amz-copy-source-if-match: CopySourceIfMatch
x-amz-copy-source-if-modified-since: CopySourceIfModifiedSince
x-amz-copy-source-if-none-match: CopySourceIfNoneMatch
x-amz-copy-source-if-unmodified-since: CopySourceIfUnmodifiedSince
x-amz-copy-source-range: CopySourceRange
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key: SSECustomerKey
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-copy-source-server-side-encryption-customer-algorithm: CopySourceSSECustomerAlgorithm
x-amz-copy-source-server-side-encryption-customer-key: CopySourceSSECustomerKey
x-amz-copy-source-server-side-encryption-customer-key-MD5: CopySourceSSECustomerKeyMD5
x-amz-request-payer: RequestPayer
x-amz-expected-bucket-owner: ExpectedBucketOwner
x-amz-source-expected-bucket-owner: ExpectedSourceBucketOwner
```

## Sample Request

```
PUT /ExampleBucket/newobject?partNumber=2&uploadId=VCVsb2FkIElEIGZvcjBlbZZpbmcncyBteS1tb3ZpZS5tMnRzIHVwbG9hZSR
HTTP/1.1
Host: 192.168.111.111:8080
Date: Mon, 11 Apr 2011 20:34:56 GMT
x-amz-copy-source: /source-bucket/sourceobject
x-amz-copy-source-range: bytes=500-6291456
Authorization: authorization string
```

## URI request Parameters

Parameter Name	Description	Required
Bucket	대상 bucket 명	Yes
Key	Multipart Upload가 시작된 개체 키입니다.	Yes
partNumber	업로드 중인 partNumber입니다. 이것은 1에서 10,000 사이의 양의 정수입니다.	
uploadId	Multipart Upload를 식별하는 업로드 ID입니다.	
x-amz-copy-source	복사 작업의 원본 개체를 지정합니다.	
x-amz-copy-source-if-match	해당 엔티티 태그(ETAG)가 지정된 태그와 일치하는 경우 개체를 복사합니다.	
x-amz-copy-source-if-modified-since	지정된 시간 이후 수정한 경우 개체를 복사합니다.	
x-amz-copy-source-if-none-match	해당 엔티티 태그(ETAG)가 지정한 ETAG와 다른 경우 개체를 복사합니다.	
x-amz-copy-source-if-unmodified-since	지정한 시간 이후 개체를 수정하지 않은 경우 개체를 복사합니다.	



Parameter Name	Description	Required
x-amz-copy-source-range	소스 개체에서 복사할 바이트 범위입니다.	
x-amz-copy-source-server-side-encryption-customer-algorithm	원본 개체의 암호를 해독할 때 사용할 알고리즘(예: AES256)을 지정합니다.	
x-amz-copy-source-server-side-encryption-customer-key	소스 개체의 암호를 해독하는 데 사용할 Amazon S3에 대해 고객이 제공한 암호화 키를 지정합니다. 이 헤더에 제공된 암호화 키는 원본 개체를 만들 때 사용된 암호화 키여야 합니다.	
x-amz-copy-source-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 128비트 MD5 다이제스트를 지정합니다. Amazon S3는 메시지 무결성 검사에 이 헤더를 사용하여 암호화 키가 오류 없이 전송되었는지 확인합니다.	
x-amz-expected-bucket-owner	예상 대상 버킷 소유자의 계정 ID입니다. 대상 버킷을 다른 계정에서 소유하는 경우 HTTP 403(액세스 거부) 오류와 함께 요청이 실패합니다.	
x-amz-request-payer	요청자가 요청에 대해 비용이 부과될 것임을 알고 있음을 확인합니다.	
x-amz-server-side-encryption-customer-algorithm	개체를 암호화할 때 사용할 알고리즘(예: AES256)을 지정합니다.	
x-amz-server-side-encryption-customer-key	데이터 암호화에 사용할 S3에 대해 고객이 제공한 암호화 키를 지정합니다.	
x-amz-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 128비트 MD5 다이제스트를 지정합니다.	

## Request Body

## Response Syntax

```
HTTP/1.1 200
x-amz-copy-source-version-id: CopySourceVersionId
x-amz-server-side-encryption: ServerSideEncryption
x-amz-server-side-encryption-customer-algorithm: SSECustomerAlgorithm
x-amz-server-side-encryption-customer-key-MD5: SSECustomerKeyMD5
x-amz-server-side-encryption-aws-kms-key-id: SSEKMSKeyId
x-amz-server-side-encryption-bucket-key-enabled: BucketKeyEnabled
x-amz-request-charged: RequestCharged
<?xml version="1.0" encoding="UTF-8"?>
<CopyPartResult>
  <ETag>string</ETag>
  <LastModified>timestamp</LastModified>
</CopyPartResult>
```

## Response Elements

Parameter Name	Description	Required
x-amz-copy-source-version-id	대상 버킷에 복사된 개체의 버전입니다.	
x-amz-request-charged	존재하는 경우 요청자가 요청에 대해 성공적으로 청구되었음을 나타냅니다. Values : requester	
x-amz-server-side-encryption	Amazon S3에 이 개체를 저장할 때 사용되는 서버측 암호화 알고리즘 Values : AES256 or aws:kms	
x-amz-server-side-encryption-aws-kms-key-id	있는 경우 개체에 사용된 AWS KMS(AWS Key Management Service) 대칭 고객 관리 키의 ID를 지정합니다.	
x-amz-server-side-encryption-bucket-key-enabled	복사된 개체가 AWS KMS(SSE-KMS)와의 서버측 암호화에 S3 버킷 키를 사용하는지 여부를 나타냅니다.	

Parameter Name	Description	Required
x-amz-server-side-encryption-customer-algorithm	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 사용된 암호화 알고리즘을 확인하는 헤더가 포함됩니다.	
x-amz-server-side-encryption-customer-key-MD5	고객이 제공한 암호화 키를 사용하여 서버측 암호화를 요청한 경우 응답에 이 헤더가 포함되어 고객이 제공한 암호화 키에 대한 왕복 메시지 무결성 확인을 제공합니다.	
x-amz-version-id	새로 만든 복사본의 버전 ID입니다.	
CopyPartResult	stringtimestamp	Yes

## Sample Response

```

HTTP/1.1 200 OK
x-amz-id-2: Vvag1LuByRx9e6j50nimru9p04ZVKnJ2Qz7/C1NPcfTWAtRPfTa0Fg==
x-amz-request-id: 656c76696e6727732072657175657374
Date: Mon, 11 Apr 2011 20:34:56 GMT
Server: KSAN-GW

<CopyPartResult>
  <LastModified>2011-04-11T20:34:56.000Z</LastModified>
  <ETag>"9b2cf535f27731c974343645a3985328"</ETag>
</CopyPartResult>

```

## HTTP Status Code

Status Code	Description
200 (OK)	요청 성공
403 (Access Denied)	권한 없음
404 (No Such Bucket)	bucket이 존재하지 않음
404 (No Such uploadId)	uploadId가 존재하지 않음
500 (Server Error)	

## Processing of requests

