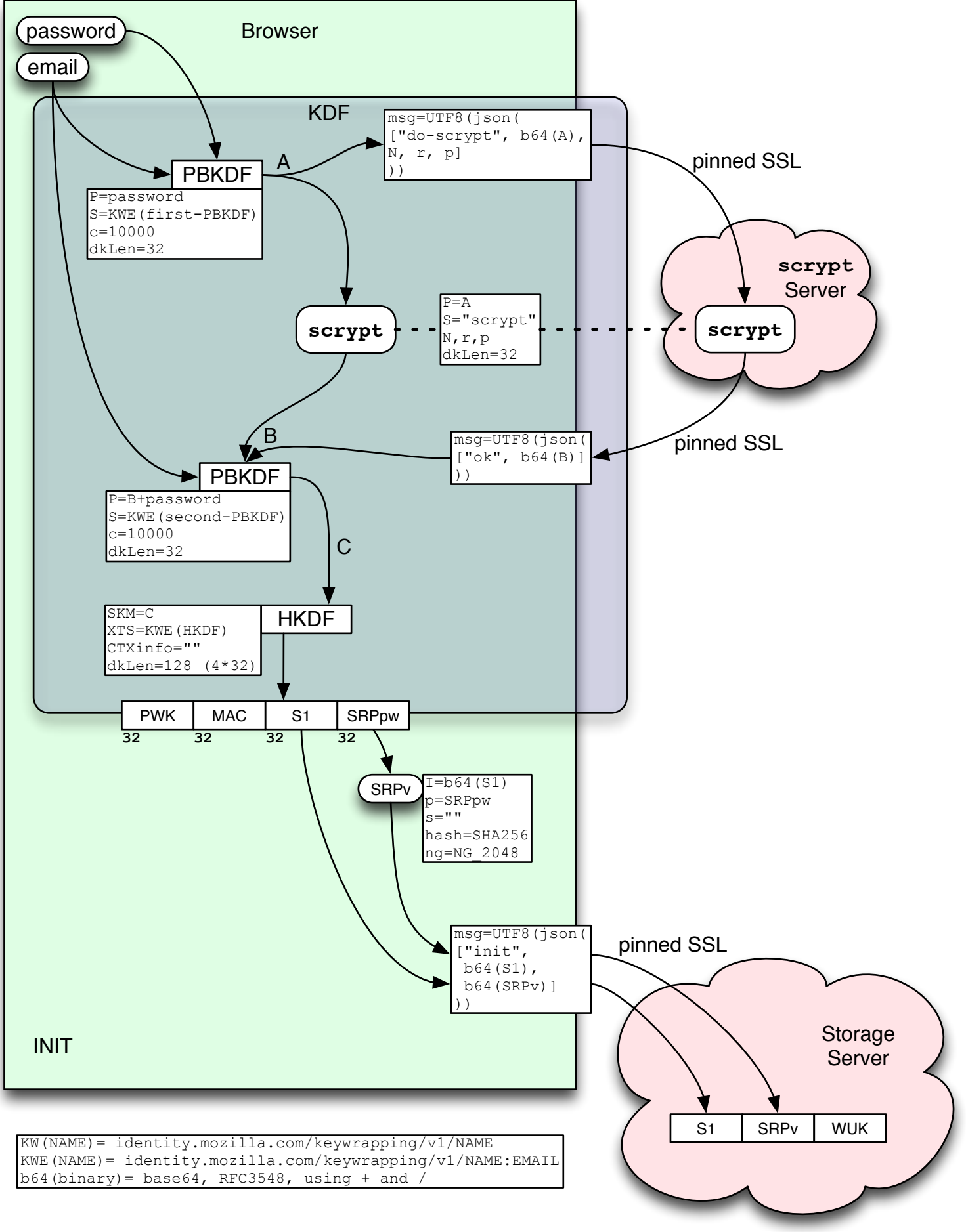
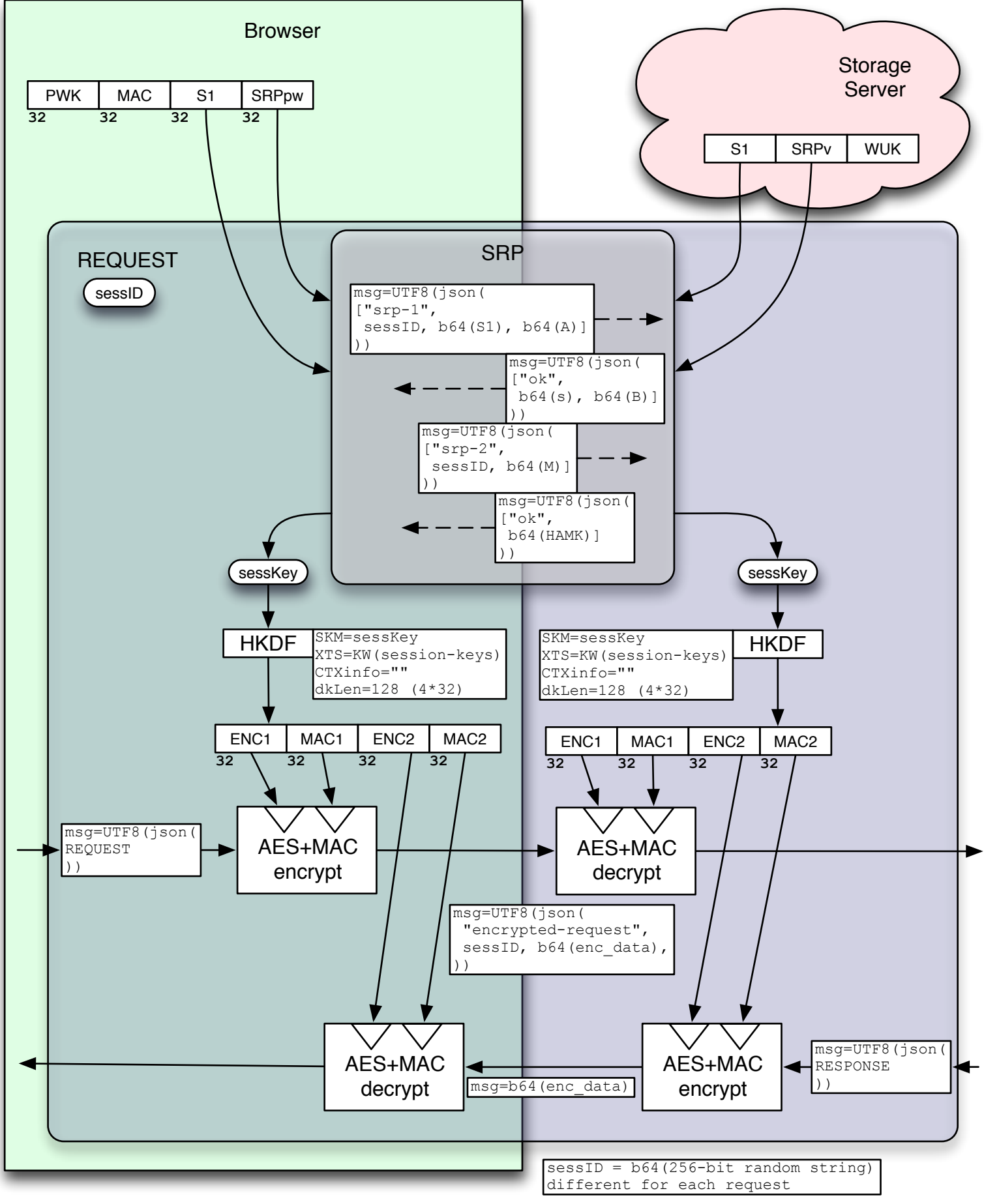
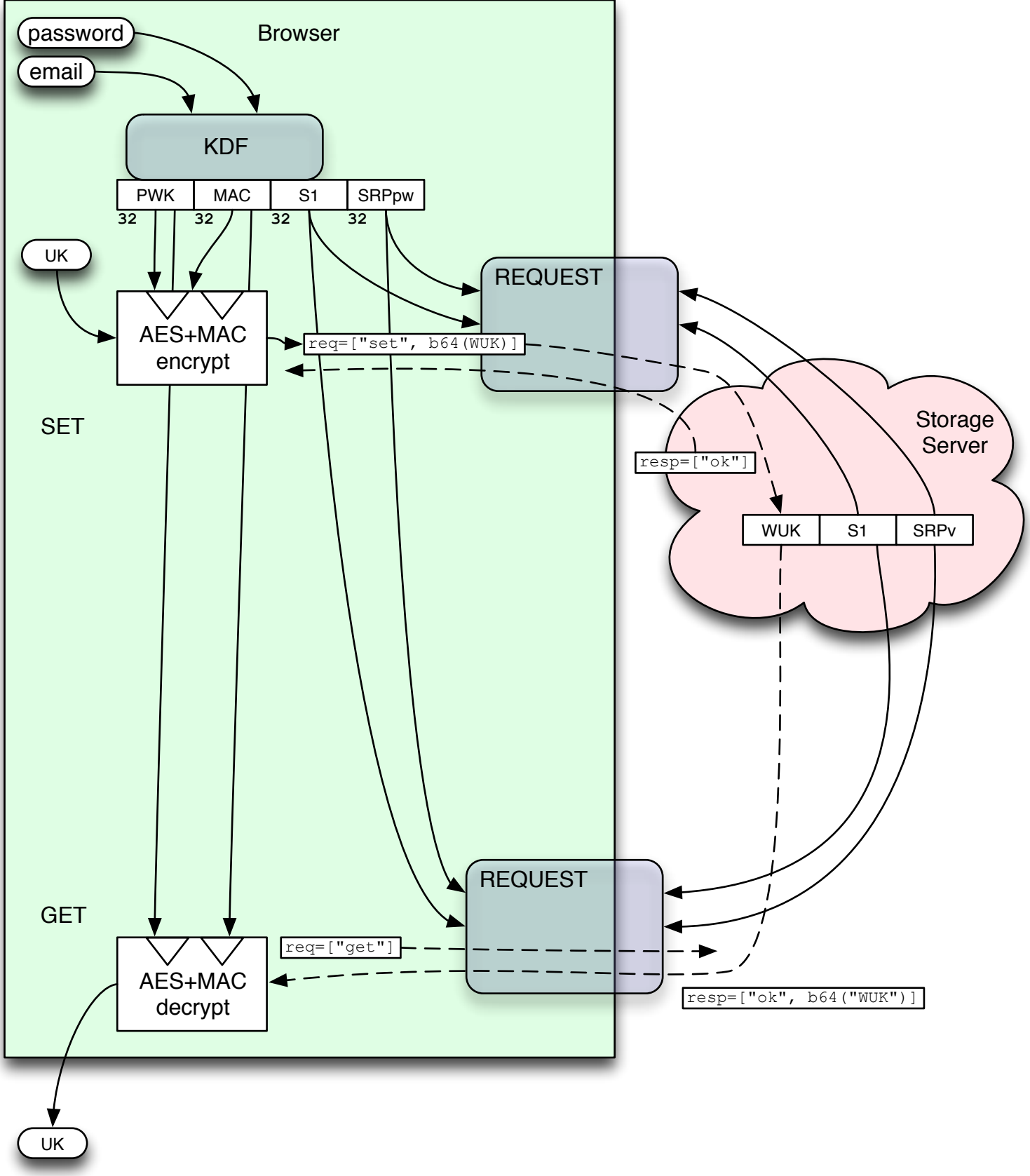
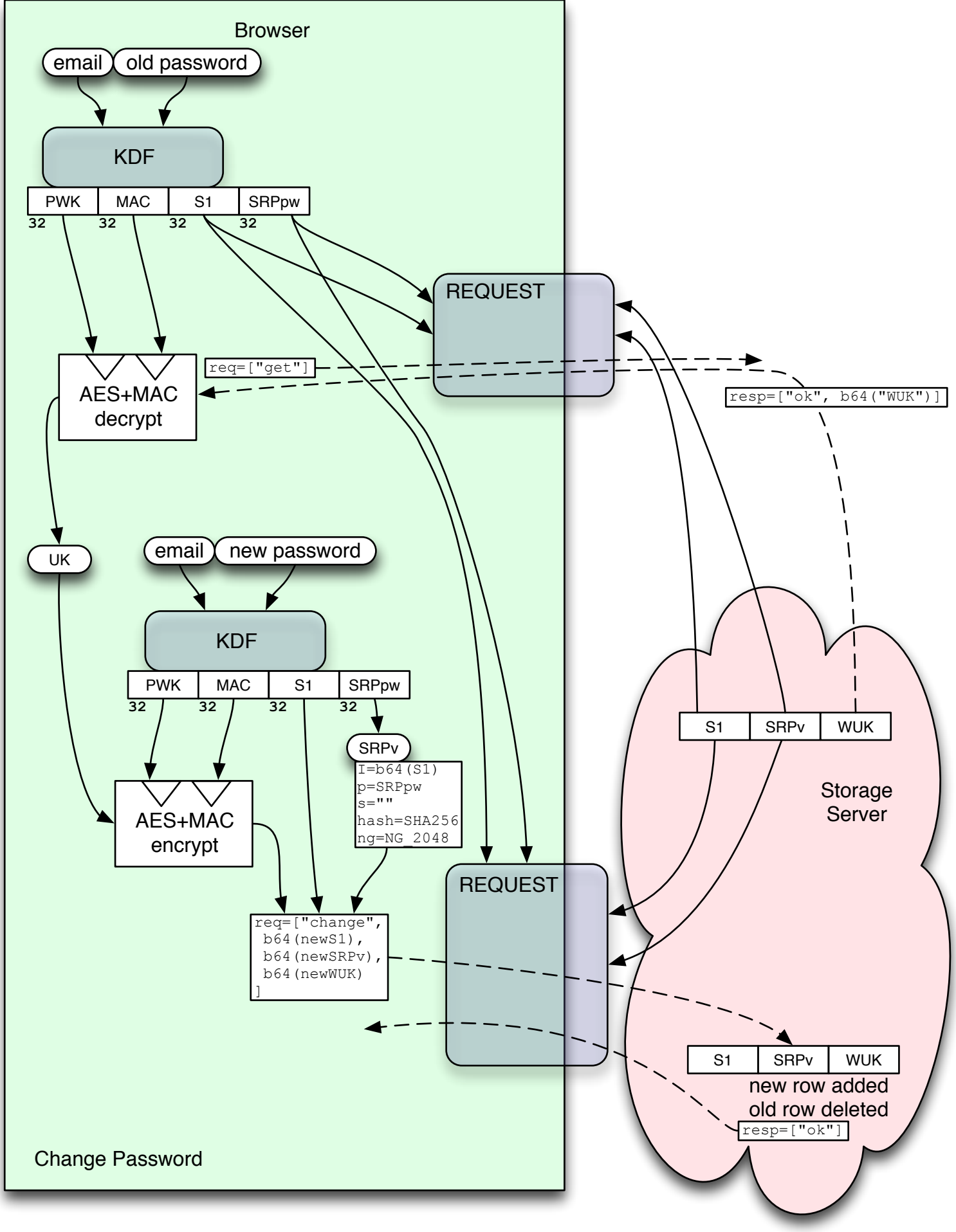
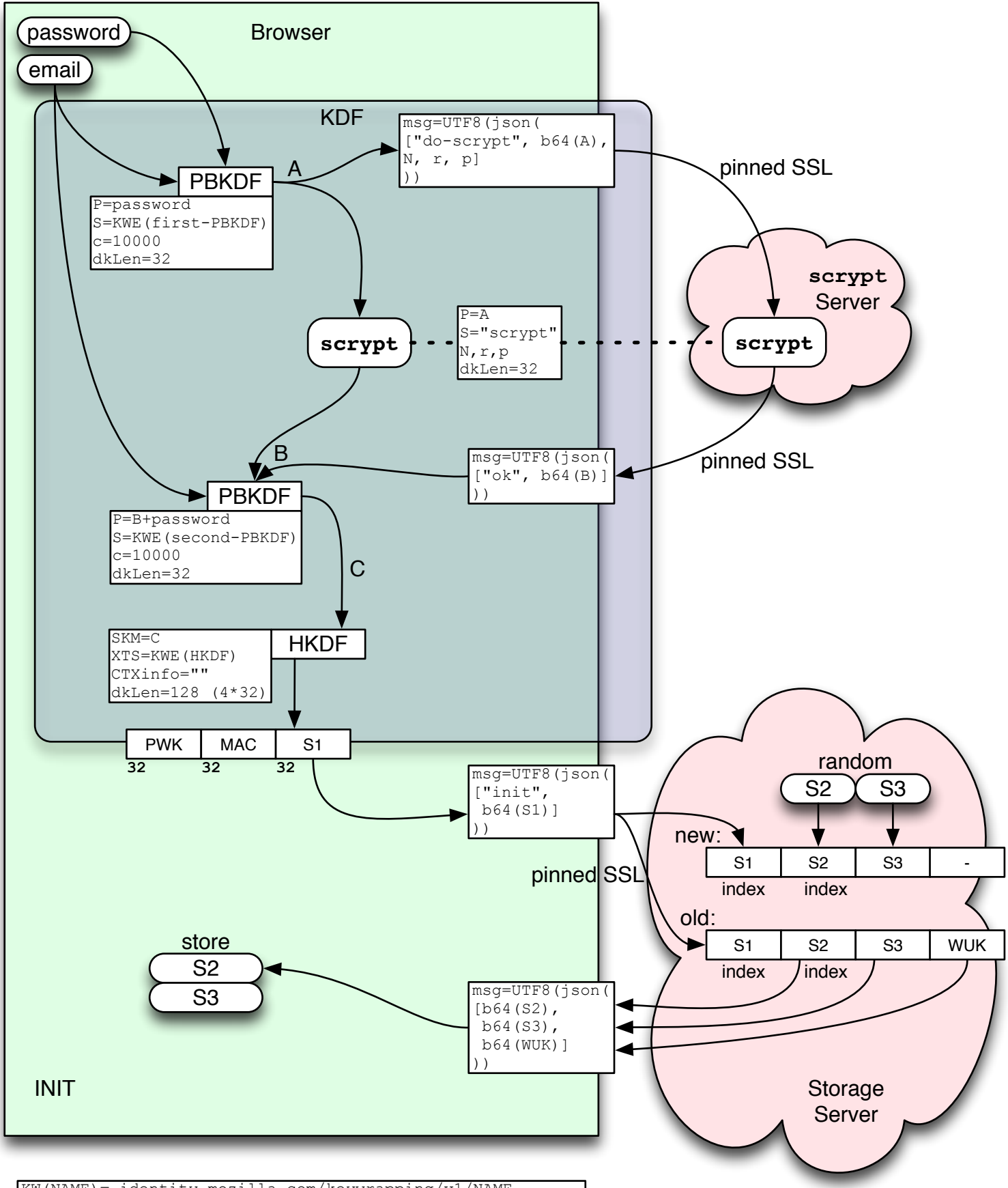

password
email
Browser
KDF
PBKDF
A
P=password
S=KWE(first-PBKDF)
c=10000
dkLen=32
msg=UTF8(json(
["do-scrypt", b64(A),
N, r, p]
))
pinned SSL
scrypt
Server
scrypt
P=A
S="scrypt"
N,r,p
dkLen=32
scrypt
B
msg=UTF8(json(
["ok", b64(B)]
))
pinned SSL
PBKDF
P=B+password
S=KWE(second-PBKDF)
c=10000
dkLen=32
C
SKM=C
XTS=KWE(HKDF)
CTXinfo=""
dkLen=128 (4*32)
HKDF
PWK
MAC
S1
SRPpw
32
32
32
32
SRPv
I=b64(S1)
p=SRPpw
s=""
hash=SHA256
ng=NG_2048
msg=UTF8(json(
["init",
b64(S1),
b64(SRPv)]
))
pinned SSL
INIT
Storage
Server
S1
SRPv
WUK
KW(NAME)= identity.mozilla.com/keywrapping/v1/NAME
KWE(NAME)= identity.mozilla.com/keywrapping/v1/NAME:EMAIL
b64(binary)= base64, RFC3548, using + and /

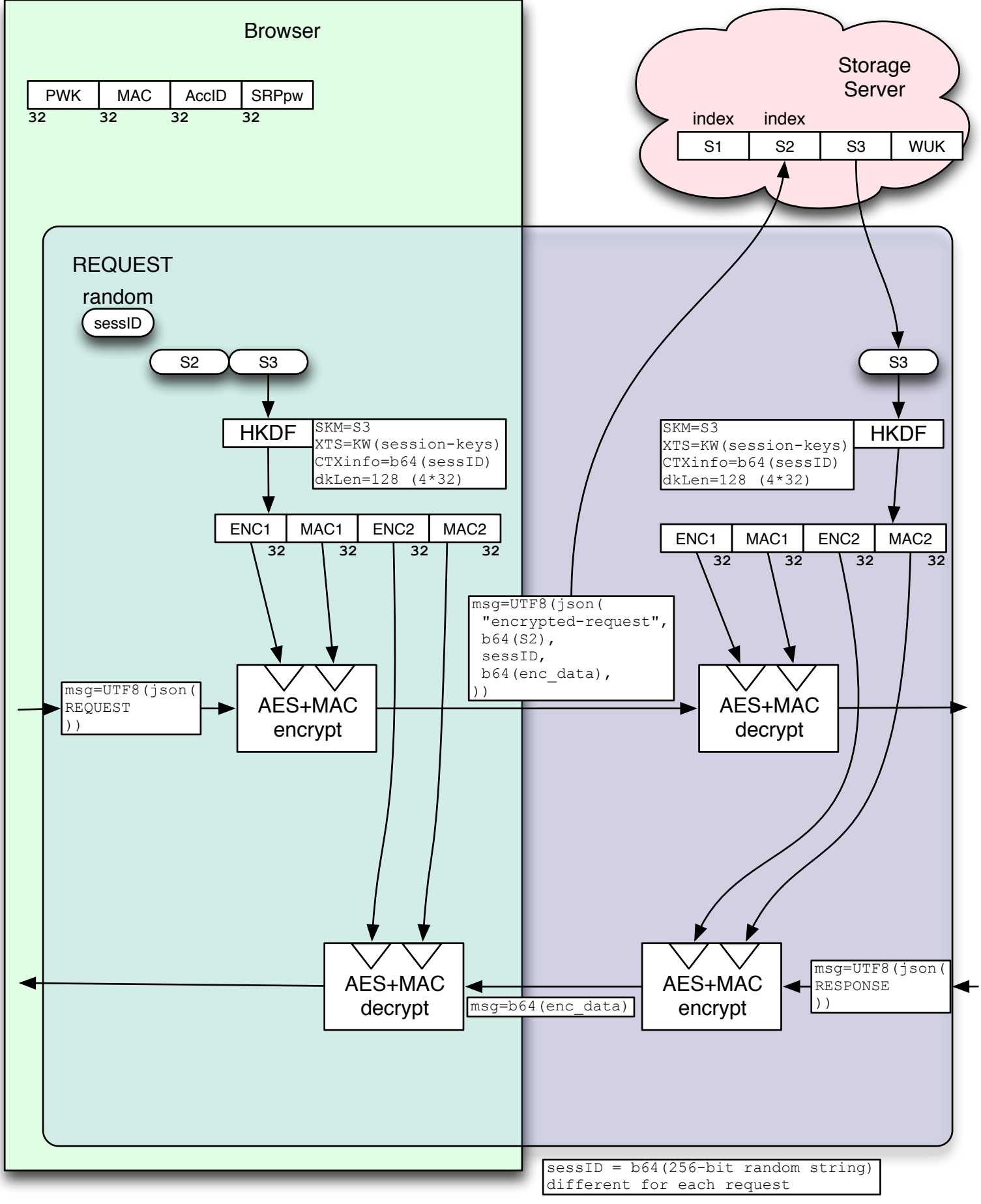

Browser
PWK
MAC
S1
SRPpw
32
32
32
32
Storage Server
S1
SRPv
WUK
REQUEST
sessID
SRP
msg=UTF8(json(
["srp-1",
sessID, b64(S1), b64(A)]
))
msg=UTF8(json(
["ok",
b64(s), b64(B)]
))
msg=UTF8(json(
["srp-2",
sessID, b64(M)]
))
msg=UTF8(json(
["ok",
b64(HAMK)]
))
sessKey
sessKey
HKDF
SKM=sessKey
XTS=KW(session-keys)
CTXinfo=""
dkLen=128 (4*32)
SKM=sessKey
XTS=KW(session-keys)
CTXinfo=""
dkLen=128 (4*32)
HKDF
ENC1
MAC1
ENC2
MAC2
32
32
32
32
ENC1
MAC1
ENC2
MAC2
32
32
32
32
msg=UTF8(json(
REQUEST
))
AES+MAC
encrypt
AES+MAC
decrypt
msg=UTF8(json(
"encrypted-request",
sessID, b64(enc_data),
))
AES+MAC
decrypt
msg=b64(enc_data)
AES+MAC
encrypt
msg=UTF8(json(
RESPONSE
))
sessID = b64(256-bit random string)
different for each request

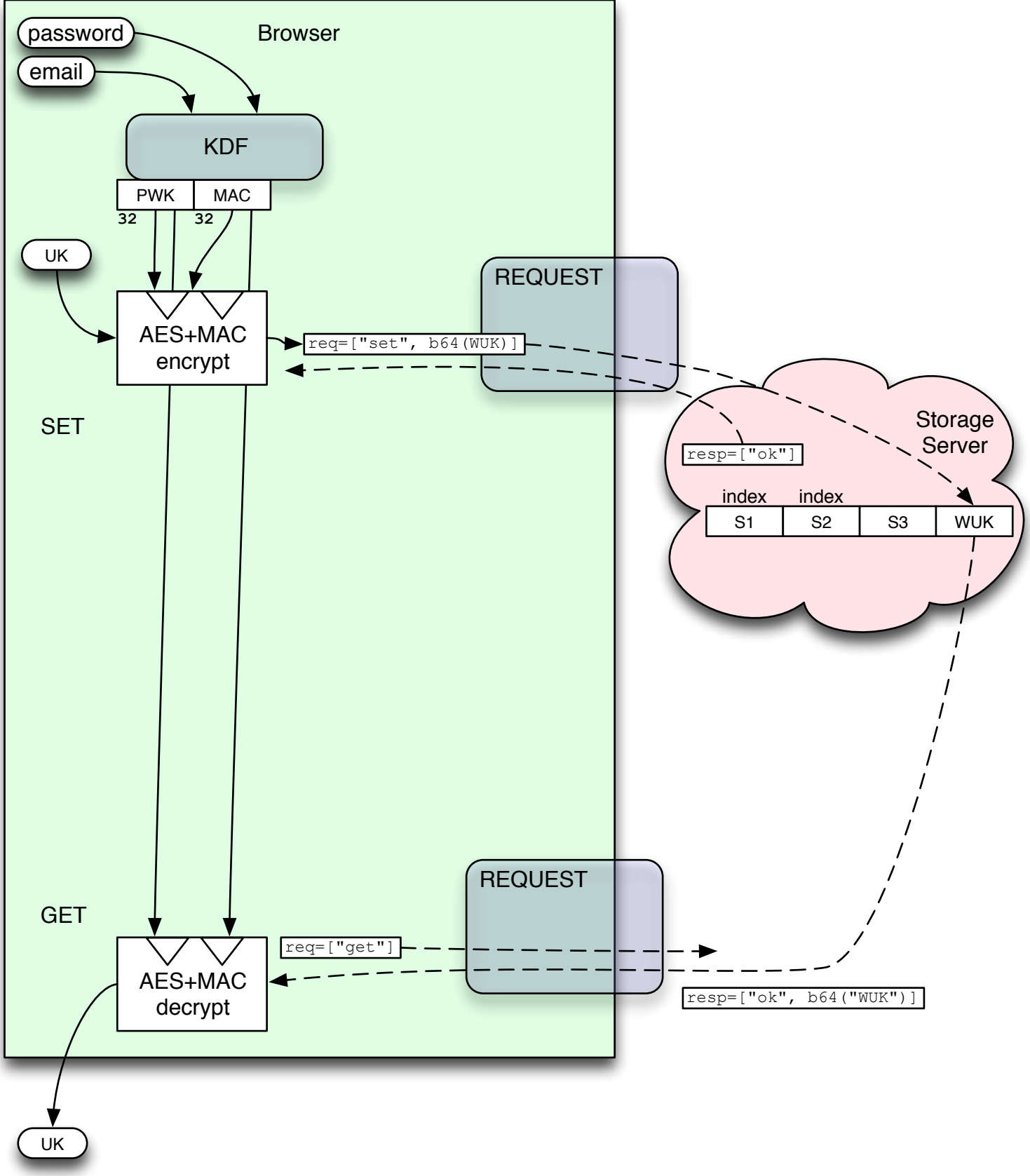

password
email
Browser
KDF
PWK
MAC
S1
SRPpw
32
32
32
32
UK
AES+MAC
encrypt
REQUEST
req=["set", b64(WUK)]
SET
Storage
Server
resp=["ok"]
WUK
S1
SRPv
REQUEST
GET
req=["get"]
AES+MAC
decrypt
resp=["ok", b64("WUK")]
UK

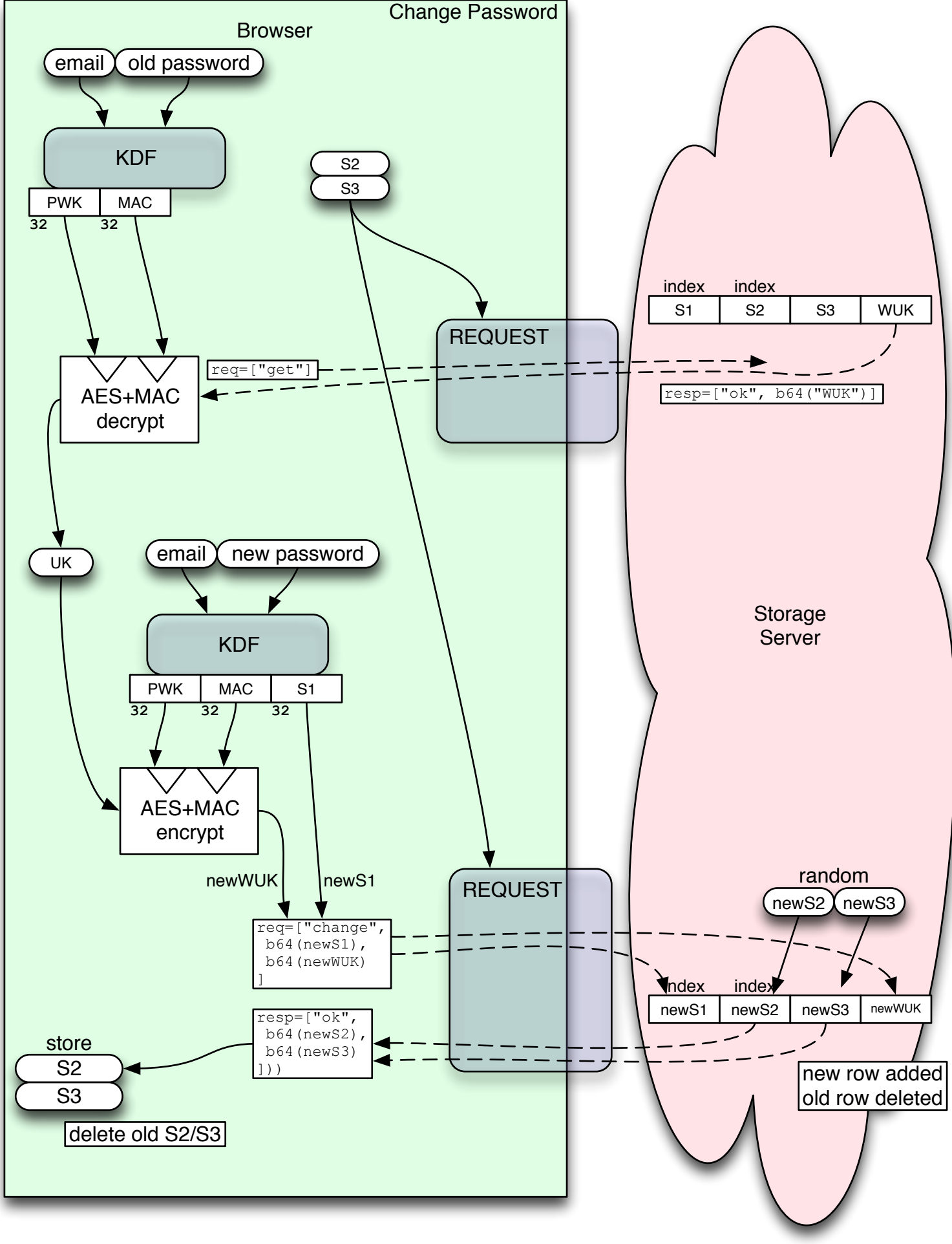

Browser
email
old password
KDF
PWK
MAC
S1
SRPpw
32
32
32
32
REQUEST
AES+MAC
decrypt
req=["get"]
resp=["ok", b64("WUK")]
UK
email
new password
KDF
PWK
MAC
S1
SRPpw
32
32
32
32
SRPv
I=b64(S1)
p=SRPpw
s=""
hash=SHA256
ng=NG_2048
AES+MAC
encrypt
REQUEST
req=["change",
b64(newS1),
b64(newSRPv),
b64(newWUK)
]
S1
SRPv
WUK
Storage
Server
S1
SRPv
WUK
new row added
old row deleted
resp=["ok"]
Change Password




```
KW(NAME)= identity.mozilla.com/keywrapping/v1/NAME
KWE(NAME)= identity.mozilla.com/keywrapping/v1/NAME:EMAIL
b64(binary)= base64, RFC3548, using + and /
```


Browser
PWK
MAC
AccID
SRPpw
32
32
32
32
Storage Server
index
index
S1
S2
S3
WUK
REQUEST
random
sessID
S2
S3
S3
HKDF
SKM=S3
XTS=KW(session-keys)
CTXinfo=b64(sessID)
dkLen=128 (4*32)
SKM=S3
XTS=KW(session-keys)
CTXinfo=b64(sessID)
dkLen=128 (4*32)
HKDF
ENC1
MAC1
ENC2
MAC2
32
32
32
32
ENC1
MAC1
ENC2
MAC2
32
32
32
32
msg=UTF8(json(
REQUEST
))
AES+MAC encrypt
msg=UTF8(json(
"encrypted-request",
b64(S2),
sessID,
b64(enc_data),
))
AES+MAC decrypt
AES+MAC decrypt
msg=b64(enc_data)
AES+MAC encrypt
msg=UTF8(json(
RESPONSE
))
sessID = b64(256-bit random string)
different for each request


password
email
Browser
KDF
PWK
MAC
32
32
UK
AES+MAC
encrypt
SET
REQUEST
req=["set", b64(WUK)]
resp=["ok"]
Storage
Server
index
index
S1
S2
S3
WUK
GET
REQUEST
req=["get"]
AES+MAC
decrypt
resp=["ok", b64("WUK")]
UK


Change Password
Browser
email
old password
KDF
PWK
MAC
32
32
S2
S3
REQUEST
index
index
S1
S2
S3
WUK
req=["get"]
resp=["ok", b64("WUK")]
AES+MAC
decrypt
UK
email
new password
KDF
PWK
MAC
S1
32
32
32
Storage
Server
AES+MAC
encrypt
newWUK
newS1
REQUEST
random
newS2
newS3
req=["change",
b64(newS1),
b64(newWUK)
]
index
index
newS1
newS2
newS3
newWUK
resp=["ok",
b64(newS2),
b64(newS3)
]))
store
S2
S3
new row added
old row deleted
delete old S2/S3