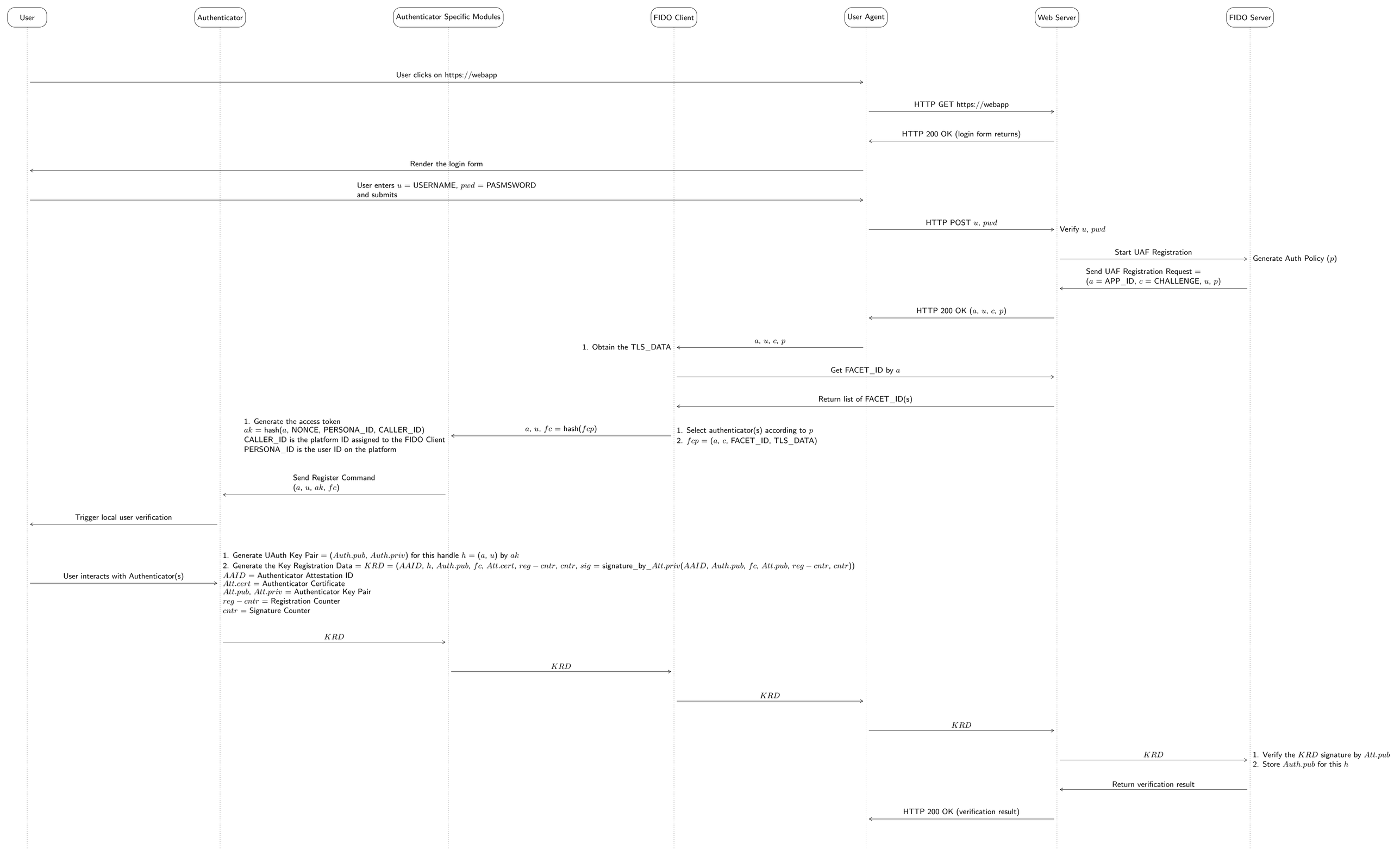
User
Authenticator
Authenticator Specific Modules
FIDO Client
User Agent
Web Server
FIDO Server
User clicks on https://webapp
HTTP GET https://webapp
HTTP 200 OK (login form returns)
Render the login form
User enters u = USERNAME, pwd = PASMSWORD
and submits
HTTP POST u, pwd
Verify u, pwd
Start UAF Registration
Generate Auth Policy (p)
Send UAF Registration Request =
(a = APP_ID, c = CHALLENGE, u, p)
HTTP 200 OK (a, u, c, p)
a, u, c, p
1. Obtain the TLS_DATA
Get FACET_ID by a
Return list of FACET_ID(s)
1. Generate the access token
ak = hash(a, NONCE, PERSONA_ID, CALLER_ID)
CALLER_ID is the platform ID assigned to the FIDO Client
PERSONA_ID is the user ID on the platform
a, u, fc = hash(fcp)
1. Select authenticator(s) according to p
2. fcp = (a, c, FACET_ID, TLS_DATA)
Send Register Command
(a, u, ak, fc)
Trigger local user verification
1. Generate UAuth Key Pair = ($Auth.pub$, $Auth.priv$) for this handle h = (a, u) by ak
2. Generate the Key Registration Data = KRD = ($AAID$, h, $Auth.pub$, fc, $Att.cert$, $reg-cntr$, $cntr$, sig = signature_by_$Att.priv$($AAID$, $Auth.pub$, fc, $Att.pub$, $reg-cntr$, $cntr$))
User interacts with Authenticator(s)
$AAID$ = Authenticator Attestation ID
$Att.cert$ = Authenticator Certificate
$Att.pub$, $Att.priv$ = Authenticator Key Pair
$reg-cntr$ = Registration Counter
$cntr$ = Signature Counter
KRD
KRD
KRD
KRD
KRD
1. Verify the KRD signature by $Att.pub$
2. Store $Auth.pub$ for this h
Return verification result
HTTP 200 OK (verification result)