PAKE - Password Authenticated Key Exchange

Prof Bill Buchanan, The Cyber Academy

http://asecuritysite.com
The problem with passwords, hashing and salt

User: alice
Password: qwerty

Hash("123456" + "6Jda0")
Hash("password" + "6Jda0")
Hash("qwerty" + "6Jda0")
SRP

Alice

“mysecret”

Bob

Stored secret

SRP

encryption key

encryption key
SPAKE2 (Password-Authenticated Key Exchange)

Alice

Bob

“mysecret”

SPAKE2

encryption key

SPAKE2

encryption key
SPAKE2 (Password-Authenticated Key Exchange)

Alice

Bob

“mysecret”

SPAKE2

encryption key

SPAKE2

encryption key

“mysecret”
SPAKE2 (Password-Authenticated Key Exchange)

\[ X = xG \text{ and } T = wM + X \]

The value of \( T \) is sent to Bob. Bob also creates \( w \), and picks a random number \( (y) \) from 0 to \( p \) (a prime number). He then calculates:

\[ Y = yG \text{ and } S = wN + Y \]

Bob sends the value of \( Y \) to Alice.

Alice calculates \( K(Alice) = x(S - wN) \), and Bob calculates \( K(Bob) = y(T - wM) \). These values are basically:

\[ K(Bob) = x(S - wN) = x(wN + Y - wN) = xy = xyG \]
\[ K(Alice) = y(T - wM) = y(wM + X - wN) = yX = xyG \]
Proving Password

"mysecret"

Hash(password,salt)

Delete after use

"mysecret"

Hash(password,salt)

Secret key (or validation)
Client secret key

Registration of password

server public key

password

salt

Alice

Bob
$K = \text{PasswordHash}($password$,$salt2$)$

$C = \text{Encrypt}(K, \text{client secret key} \mid \text{server’s public key})$
PAKE - Password Authenticated Key Exchange

Prof Bill Buchanan, The Cyber Academy
http://asecuritysite.com