Gong

Author(s): Li Gong 1989
Last modified February 6, 2003

Summary: Mutual authentication protocol of two principals with a trusted server, and exchange of a new symmetric key. Uses one-way functions and no encryption.

Protocol specification (in common syntax)

```
A, B, S :      principal
Na, Nb, Ns :   number
Pa, Pb :       number
K, Ha, Hb :    number
f1 :           number, number, number, number -> number
f2 :           number, number, number, number -> number
f3 :           number, number, number, number -> number
g :            number, number, number, number -> number
xor :          number,number -> number

alias K = f1(Ns,Na,B,Pa)
alias Ha = f2(Ns,Na,B,Pa)
alias Hb = f3(Ns,Na,B,Pa)

1.  A  ->  B  :  A, B, Na
2.  B  ->  S  :  A, B, Na, Nb
3.  S  ->  B  :  Ns, xor(f1(Ns, Nb, A, Pb), K),
                 xor(f2(Ns, Nb, A, Pb), Ha),
                 xor(f3(Ns, Nb, A, Pb), Hb),
                 g(K, Ha, Hb, Pb)
4.  B  ->  A  :  Ns, Hb
5.  A  ->  B  :  Ha
```

Description of the protocol rules

`f1`, `f2`, `f3`, and `g` are one-way functions.

Initially, the principal `A` (resp. `B`) shares the long-term secret `Pa` (resp. `Pb`) with the server `S`.

`Na`, `Nb` and `Ns` are nonces and `Ha`, `Hb` and `K` are just aliases for resp. `f2(Ns,Na,B,Pa)`, `f3(Ns,Na,B,Pa)`, and `f1(Ns,Na,B,Pa)`.

We assume commutativity and associativity for the `xor` operator,

```
xor(x, y) = xor(y, x)
xor(x, xor(y, z)) = xor(xor(x, y), z)
```

as well as the xor axioms:

```
xor(x, 0) = x
xor(x, x) = 0
```

Hence, the principal `B` can extract `Ha`, `Hb` and `K` from the message `3`, and check them using the checksum `g(K, Ha, Hb, Pb)`.

Knowing `Pa`, the principal `A` can construct `Ha`, `Hb` and `K` once he has received `Ns` in message `4`, he can check the check value for `Hb` he has just computed against the one sent by `B` in message `4` and send the last message.

Requirements

The protocol must guaranty the secrecy of `K`: in every session, the value of `K` must be known only by the participants playing the roles of `A`, `B` and `S` in that session.

The protocol must also ensure mutual authentication of `A` and `B`.

References

[Gon89]

Citations

[Gon89] Li Gong. Using one-way functions for authentication. *Computer Communication Review*, 19(5):8–11, october 1989.