Protocols

Steven Allen

April 18, 2013

1 Analysis Variables

R Number of resources

G Number of groups per person

F Number of friends per person

S Number of services

M Average members per group

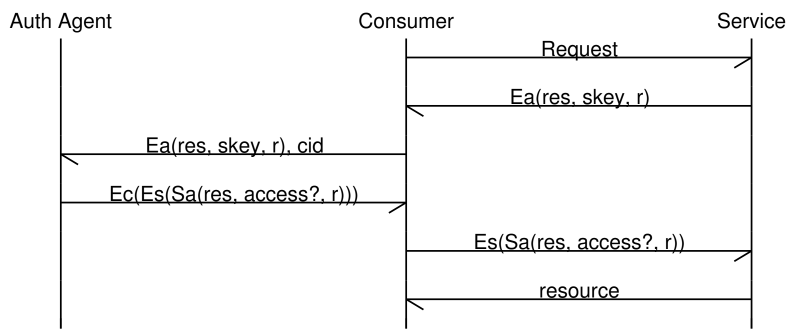
r Request rate

g Group change rate

p Resource post rate

2 Obvious PK

2.1 Protocol



2.2 Costs

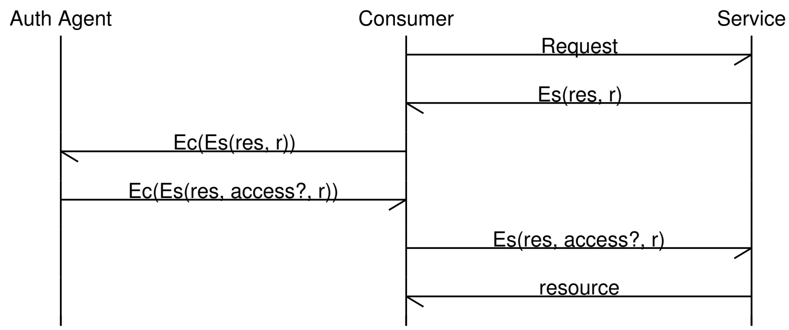
Op	Auth Agent	Consumer	Service
Encryption	$\Theta(r)$	$\Theta(r)$	$\Theta(r)$
Storage	$\Theta(RG + RS)$	$\Theta(1)$	$\Theta(R)$
Transfer	$\Theta(r)$	$\Theta(r)$	$\Theta(r)$

2.3 Analysis

Perfect privacy.

3 Obvious Shared Secret

3.1 Protocol



3.2 Costs

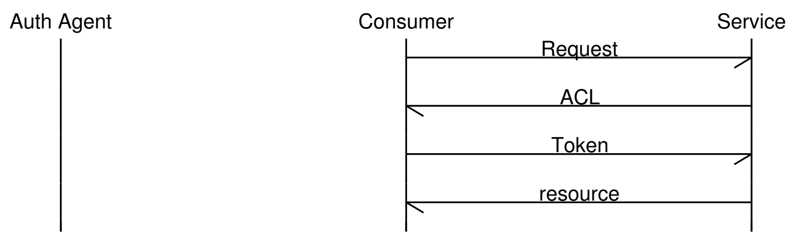
Op	Auth Agent	Consumer	Service
Encryption	$\Theta(r)$	$\Theta(r)$	$\Theta(r)$
Storage	$\Theta(RG + RS)$	$\Theta(F)$	$\Theta(R)$
Transfer	$\Theta(r)$	$\Theta(r)$	$\Theta(r)$

3.3 Analysis

Perfect privacy.

4 Public Key ACL (Basic)

4.1 Protocol



4.2 Costs

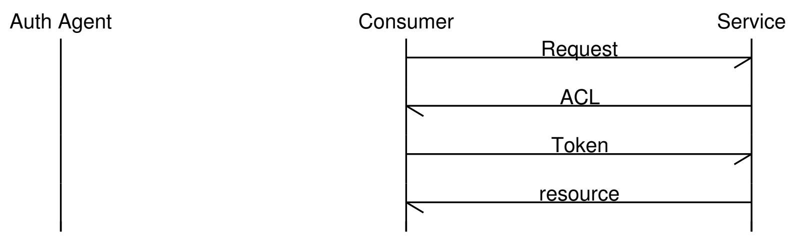
Op	Auth Agent	Consumer	Service
Encryption	$\Theta(g)$	$\Theta(rM)$	0
Storage	$\Theta(GM)$	$\Theta(1)$	$\Theta(RM)$
Transfer	$\Theta(g)$	$\Theta(rM)$	$\Theta(rM)$

4.3 Analysis

The ACL is of the form: $\{E_{c_1}(t), E_{c_2}(t), ...\}$. Consumers are able to determine the size of the ACL group.

5 Public Key ACL (Per Group)

5.1 Protocol



5.2 Costs

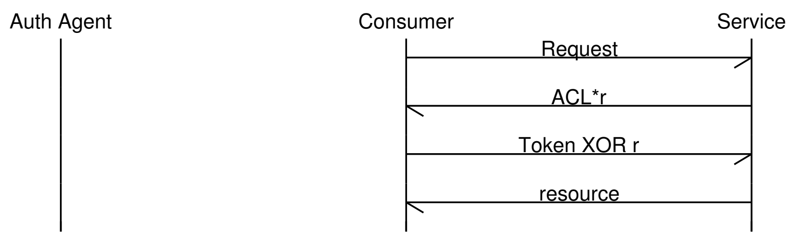
Op	Auth Agent	Consumer	Service
Encryption	$\Theta(g)$	$\Theta(rM)$	0
Storage	$\Theta(GM)$	$\Theta(1)$	$\Theta(GM)$
Transfer	$\Theta(g)$	$\Theta(rM)$	$\Theta(rM)$

5.3 Analysis

Both the consumers and the services are able to group content.

6 Public Key ACL (per group, enhanced)

6.1 Protocol



6.2 Costs

Op	Auth Agent	Consumer	Service
Encryption	$\Theta(g)$	$\Theta(rM)$	0
Storage	$\Theta(GM)$	$\Theta(1)$	$\Theta(GM)$
Transfer	$\Theta(g)$	$\Theta(rM)$	$\Theta(rM)$

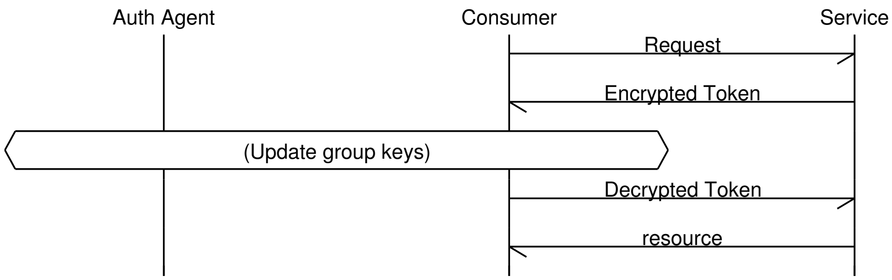
6.3 Analysis

Using Goldwasser-Micali for pk encryption, compute $r * \text{ACL} = \{E_{c_1}(t \oplus r), E_{c_2}(t \oplus r), ...\}$. This allows us to hide the group from the consumer.

With this encryption scheme, the consumers can't group content but the server still can.

7 El-Gamal Variant

7.1 Protocol



7.2 Costs

Op	Auth Agent	Consumer	Service
Encryption	$\Theta(p+g)$	$\Theta(r)$	$\Theta(r+p)$
Storage	$\Theta(GM)$	$\Theta(GF)$	$\Theta(R)$
Transfer	$\Theta(g+p)$	$\Theta(r)$	$\Theta(r+p)$

7.3 Description

This scheme uses a variant of ElGamal that allows for multiple uncorrelatable (to be proven) public keys per private key.

A loose description of the encryption scheme follows (there are a few other requirements but are not necessary to get the gist of it).

The consumer key consists of two parameters: x, w. The service keys are defined as g^{ax}, g^{a-w} where a is chosen randomly and uniquely per resource posted. Under this scheme, encryption is defined as $E_a(m) = (g^r, g^{ar-wr}, mg^{axr}) = (j, k, l)$ and decryption is defined as $D_a(j, k, l) = l/(kj^w)^x$.

The token that the service sends the consumer is $E_a(m)$, and the decrypted token is m.

To prevent man-in-the-middle attacks, m should encode information about the server requesting authentication (IP address, a hash of their SSL key, etc).

Also, as ElGamal (and therefore this scheme) is malleable, the message will have to be properly padded to ensure that it isn't modified.

8 Privacy Analysis

In this system, the server is unable to identify users, or even which groups guard a piece of content. However, users are able to tell if they are in the same group by comparing private group keys. Unfortunately, I was unable to come up with a way around this.