AARHUS UNIVERSITY - COMPUTER SCIENCE

Security - Handin 2

Peter Burgaard - 201209175

April 26, 2015

We are given two methods of safe database communication which should comply with the firms security policy:

1. When a request for information arrives, D should be able to determine which user sent the request.
2. A user should not be able to get information about which data other users asked for.

The two methods are as follows:

$$E_{pk_D}(R), S_{sk_A}(E_{pkd}(R)), A \tag{1}$$

$$E_{pk_d}(R, S_{sk_A}), A \tag{2}$$

Then we're asked if; "One of these solutions fails to satisfy the security policy". If an malicious employee wanted to get information about which data other users were asking for and, the company had installed protocol (1), all they had to do was retrieve the request they wanted information from, and encrypt the $E_{pk_D}(R)$ with their own S_{sk_a}, and send a request with the encrypted request, the newly self-signed encrypted request, and their information A, and they'd receive the requested information from the database. This is against the security policy article 2 about, users not being able to get information about data other users have received from the database.

It's also obvious that if the malicious employee just wanted to mess with the system, they could just put their own information A in a random intercepted request, which would break the security policy article 1 about the database being deterministic in the identification of the data requesting user, since A and the signing S_{sk_A} isn't matching anymore, the data D doesn't know which user is asking for data.

If the data requesting method (2) is used, we're still able to to breake security policy article 1, but now we can't make a fake request, and get the other users that.

Generaly we can say method (2) is the better in the way that it provides more secure authenticity, since the database can be safely assume if it recieves a correct request, where the signing and A matches, it is indeed the correct user asking for data. If A and S_{sk_A} doesn't match the database could deny a such request, and the system would be safe with respect to article 1, but it still wouldn't be safe with respect to 1.

Conclussion if, you're going to send composite information, encrypt it all at once with the same key, not in parts.