IKE message. HDR_j^* indicates that all payloads following HDR_j are encrypted. SA_i and SA_r are security association payloads. KE_i and KE_r are key exchange payloads. N_i and N_r are nonce payloads. ID_{ii} and ID_{ir} are identification payloads. (The subscripts i and r in these payloads represent the initiator and the responder, respectively.) *HASH_I* and *HASH_R* are authenticators generated by the initiator and the responder, respectively.

At the end of step 4, the newly shared secret SKEYID can be derived as follows:

$$SKEYID = prf(pre\text{-}shared\text{-}key,\ N_i\_b \mid N_r\_b)$$

where *prf* is a keyed pseudorandom function, *pre-shared-key* is a key pre-shared between the initiator and the responder, and N_i_b and N_r_b are nonce payloads excluding their generic payload heads.

The authenticators *HASH_I* and *HASH_R* are defined as follows:

$$HASH\_I = prf(SKEYID,\ g^{x_i}|g^{x_r}|CKY\text{-}I|CKY\text{-}R|SA_i\_b|ID_{ii}\_b)$$
$$HASH\_R = prf(SKEYID,\ g^{x_r}|g^{x_i}|CKY\text{-}R|CKY\text{-}I|SA_i\_b|ID_{ir}\_b)$$

where g^{x_i} and g^{x_r} are Diffie-Hellman public values included in key exchange payloads KE_i and KE_r, respectively, *CKY-I* and *CKY-R* are cookies generated by the initiator and the responder, respectively, and included in HDR_j (j = 1, 2, ..., 6) to identify an ISAKMP security association, SA_i_b is the security association payload of the initiator excluding its generic payload head, and ID_{ii}_b and ID_{ir}_b are identification payloads excluding their generic payload heads.

The above definitions of *HASH_I* and *HASH_R* are unable to authenticate the ISAKMP security association being negotiated.

Attack: In the aforementioned protocol, the initiator presents offers for potential security association to the responder in the SA_i payload. Each offer contains a set of security attributes which includes an encryption algorithm, hash algorithm, authentication method, information about a group on which to carry out a Diffie-Hellman exchange, life type and duration of the security association. The responder will choose only one of the offers provided by the initiator, and will send it back to the initiator in the SA_r payload.

Suppose that SA_i contains two offers $T_{\#1}$ and $T_{\#2}$ supplied by the initiator. SA_r contains an offer $T_{\#1}$ chosen by the responder. Suppose A is an attacker which generates SA_a containing another offer $T_{\#2}$. An attack can be launched as follows:

1. $I \rightarrow R$: $HDR_1,\ SA_i$
2. $R \rightarrow A$: $HDR_2,\ SA_r$
2′. $A \rightarrow I$: $HDR_{2'},\ SA_a$
3. $I \rightarrow R$: $HDR_3,\ KE_i,\ N_i$
4. $R \rightarrow I$: $HDR_4,\ KE_r,\ N_r$
5. $I \rightarrow R$: $HDR_5^*,\ ID_{ii},\ HASH\_I$
6. $R \rightarrow I$: $HDR_6^*,\ ID_{ir},\ HASH\_R$

The attacker can intercept SA_r being sent from the responder to the initiator at step 2, then forward SA_a to the initiator at step 2′. At the end of the protocol, the initiator will believe that an ISAKMP security association with an offer $T_{\#2}$ has been established while the responder will believe that an ISAKMP security association with an offer $T_{\#1}$ has been established. Neither the initiator nor the responder will be aware of such an attack. This is caused by the flawed definition of authenticators *HASH_I* and *HASH_R* in which SA_i, rather than SA_r, is used to authenticate the security association being established.

The attack may not be successful if the security attributes such as the encryption algorithm and hash algorithm defined in SA_a and SA_r are different, since *HASH_I* and *HASH_R* are generated and verified with these algorithms. However, it should be noted that not all security attributes are related to the generation and verification of *HASH_I* and *HASH_R*. For example, if SA_a and SA_r have the same security attributes with the exception of their life type and duration, the above attack cannot be detected.

As *HASH_I* and *HASH_R* are used in every phase 1 protocol to authenticate the ISAKMP security association being negotiated, all of those protocols are vulnerable to the attack.

Amendment: To avoid the above attack, authenticators *HASH_I* and *HASH_R* should be defined as follows:

$$HASH\_I = prf(SKEYID,\ g^{x_i}|g^{x_r}|CKY\text{-}I|CKY\text{-}R|SA_r\_b|ID_{ii}\_b)$$
$$HASH\_R = prf(SKEYID,\ g^{x_r}|g^{x_i}|CKY\text{-}R|CKY\text{-}I|SA_r\_b|ID_{ir}\_b)$$

We only need to replace SA_i_b with SA_r_b in *HASH_I* and *HASH_R*. Thus the offer chosen by the responder as the security association being established will be explicitly authenticated.

As the payload SA_r of the responder contains only one of the offers chosen from the payload SA_i of the initiator, the size of SA_r will be no larger than that of SA_i. Hence, the amendment will reduce the computation overheads of *HASH_I* and *HASH_R*.

Conclusion: We have shown the weakness of and provided an amendment to IKE protocols. The amendment will not only make the protocols more robust, but also reduce their computational overheads to some extent.

15 April 1999
Electronics Letters Online No. 19990747
DOI: 10.1049/el:19990747

J. Zhou (*Kent Ridge Digital Labs, 21 Heng Mui Keng Terrace, Singapore 119613, Republic of Singapore*)

E-mail: jyzhou@krdl.org.sg

References

1 HARKINS, D., and CARREL, D.: 'The Internet key exchange (IKE)'. RFC 2409, November 1998
2 MAUGHAN, D., SCHERTLER, M., SCHNEIDER, M., and TURNER, J.: 'Internet security association and key management protocol (ISAKMP)'. RFC 2408, Nov. 1998
3 ORMAN, H.: 'The Oakley key determination protocol'. RFC 2412, November 1998
4 PIPER, D.: 'The Internet IP security domain of interpretation for ISAKMP'. RFC 2407, November 1998

Simple authenticated key agreement algorithm

Dong Hwi Seo and P. Sweeney

A password-based method is described which modifies the Diffie-Hellman key agreement protocol to provide user authentication. It is simpler than previously published schemes, prevents the man-in-the-middle attack and requires only two packets to agree on the secret session key. An optional exchange of two more packets allows the key agreement to be verified.

Introduction: Diffie-Hellman key agreement [1] is a well known method by which two authenticated parties may use an insecure channel to agree a session key for use with conventional symmetric encryption algorithms. It makes use of the difficulty of computing discrete logarithms over a finite field. The main problem of the Diffie-Hellman key exchange method is that it is vulnerable to man-in-the-middle attacks. As the Diffie-Hellman key exchange does not authenticate the participants, it is possible for a man-in-the-middle (Eve) to interpose in the line and impersonate Bob to Alice and Alice to Bob.

Adopting certificates (e.g. digital signature) into a key exchange protocol can foil man-in-the-middle attacks. A certificate from a trusted authority (certifying authority, CA) is presented to the user along with the public key to certificate the ownership of the keys. Now, Eve cannot impersonate either Alice or Bob and cannot substitute the original public keys with her own because they are signed. A public key system such as RSA can be used for this

purpose. One example of this scheme is the authenticated Diffie-Hellman key agreement protocol, or station-to-station (STS) protocol, which was developed by Diffie *et al.* [2].

As key exchange schemes with certificates require some trusted authority to verify the integrity of the received messages, the extension to a larger system may be difficult. They need a large storage for certificates and more bandwidth for the verification of the signature as the number of users increases. Furthermore, if the authority is compromised then the total system would be in danger.

Another kind of authenticated key exchange protocol, which assumes a pre-shared secret password between two users, has been suggested. The security of the system is dependent on the individuals, not on the role of a third authority. In encrypted key exchange (EKE) [3] a shared password P is used as a key to encrypt a randomly generated number. This scheme defeats man-in-the-middle attacks, as Eve has no method to disguise herself as Alice and Bob without knowing the password P. But this algorithm is complicated and is also patented, obstructing wide usage. Another example of this type of scheme is fortified key negotiation [4].

In this Letter, a new key agreement protocol is described, the simple authenticated key agreement algorithm (SAKA), based on the Diffie-Hellman protocol. It prevents man-in-the-middle attacks and the amount of generated traffic is the same as that of the Diffie-Hellman approach with only two packets required to agree on the secret session key.

SAKA: SAKA requires that Alice and Bob share a common password P before the protocol begins and uses the same public values of g and n as the original Diffie-Hellman.

(1) Alice and Bob each compute two integers Q and (Q^{-1}) mod $(n - 1)$ from the password P. Q could be computed in any predetermined way from P, provided it yields a unique value, relatively prime with $(n - 1)$, and with low probability that two different passwords will give the same value of Q. For example, Q could be the smallest such integer that is greater than a numeric representation of the password P.

(2) Alice chooses a random large integer a and sends Bob

$$X_1 = g^{aQ} \bmod n$$

(3) Bob chooses a random large integer b and sends Alice

$$Y_1 = g^{bQ} \bmod n$$

(4) Alice computes

$$Y = Y_1^{Q^{-1}} \bmod n$$
$$Key\_1 = Y^a \bmod n$$

(5) Bob computes

$$X = X_1^{Q^{-1}} \bmod n$$
$$Key\_2 = X^b \bmod n$$

Proof: If the product of two integers (q and r) is 1 modulo $(n - 1)$, where n is prime then, using Fermat's Little Theorem, it can be expressed as

$$qr = 1 \bmod (n - 1)$$
$$r = q^{-1} \bmod (n - 1) \quad (1)$$

Let

$$X = g^x \bmod n$$
$$Y = X^q \bmod n = g^{xq} \bmod b$$

then

$$Y^r \bmod n = X^{rq} \bmod n \quad (2)$$

Now rq can be expressed in the form $k(n - 1) + 1$, where k is an integer. So, eqn. 2 can be represented in the following form:

$$= X^{k(n-1)+1} \bmod n$$
$$= (X.X^{k(n-1)}) \bmod n$$
$$= ((X \bmod n) * (X^{k(n-1)} \bmod n)) \bmod n$$

Since $X = g^x \bmod n$, X is relatively prime to n, because n itself is a prime number and X is less than n. Therefore, $X^{(n-1)} \bmod n = 1$. Therefore

$$X^{rq} \bmod n = X \quad \text{if} \quad r = q^{-1} \bmod (n - 1)$$

Man-in-the-middle attack: With the original Diffie-Hellman, Eve can alter the public values such as $g^a \bmod n$ and $g^b \bmod n$ from Alice and Bob with her own values. Then Eve and Alice share one key, and Eve and Bob share another key without notice. But, with SAKA, when Eve receives ($X1 = g^{aQ} \bmod n$) in step (2), she cannot guess '$g^a \bmod n$' and Q, since the problem is combined with the discrete logarithm and a secret password. If she still tries to eavesdrop, she has to make ($g^{a*Q*} \bmod n$) and send it to Bob. If Bob tries to solve (($g^{a*Q*} \bmod n)^{Q^{-1}} \bmod n$), he will obtain a wrong value, which it is impossible for Eve to know. If Alice or Bob suspects that the subsequent messages are not being decrypted correctly, the following protocol can be invoked:

If Alice and Bob want to check the validity of the session key:

(1) Alice computes $(Key\_1)^Q \bmod n$ and sends it to Bob

(2) Bob computes $(Key\_2)^Q \bmod n$ and sends it to Alice

(3) Alice and Bob each compute the other's key by applying Q^{-1} and compare it with his/her own session key.

Eve does not know Q or Q^{-1} and therefore cannot send values that will result in Alice and Bob recomputing the same key values as before.

21 April 1999
Electronics Letters Online No: 19990724
DOI: 10.1049/el:1999724

Dong Hwi Seo and P. Sweeney (*Centre for Communication System Research, School of Electronic Engineering, Mathematics and Information Technology, University of Surrey, Guildford, GU2 5XH, United Kingdom*)

References

1 DIFFIE, W., and HELLMAN, M.E.: 'New directions in cryptography', *IEEE Trans.*, 1976, **IT-22**, pp. 644–654

2 DIFFIE, W., VAN OORSCHOT, P.C., and WIENER, M.J.: 'Authentication and authenticated key exchanges', *Des. Codes Cryptogr.*, 1992, **2**, pp. 107–125

3 BELLOVIN, S.M., and MERRITT, M.: 'Encrypted key exchange: Password-based protocols secure against dictionary attacks'. Proc. 1992 IEEE Computer Society Conf. on Research in Security and Privacy, May 1992, pp. 72–84

4 ANDERSON, R.J., and LOMAS, T.M.A.: 'Fortifying key negotiation schemes with poorly chosen passwords', *Electron. Lett.*, 1994, **30**, pp. 1040–1041

Compact polymeric wavelength division multiplexer

M.A. Cowin, M. Owen, J.D. Bainbridge, R.V. Penty and I.H. White

A compact polymeric wavelength division multiplexer based on a slab waveguide and a reflective transmission grating is presented for application within the low cost data-commmunications market. The device channel spacing is 800GHz (6.4nm), measures only 4 × 4mm in size, displays a polarisation-dependent wavelength shift of 0.1nm and crosstalk levels approaching –25dB.

Introduction: Polymeric materials hold the potential to provide a means by which the expected demand for cheap, reliable and robust components for routing and switching within future WDM networks may be met. The recent emergence of low loss, thermally stable polymers [1, 2] has led to an increase in the application of polymer technology for the fabrication of passive polymeric WDM components [3 – 5]. Polymer materials offer cheap, low temperature processing of high quality, low loss waveguide structures. They display ease of manufacture on a variety of substrates and exhibit material characteristics such as refractive index and absorption that may be engineered to suit specific applications.

A simple and compact design for a four channel polymeric wavelength division multiplexer suitable for low cost data-communications applications is presented for the first time based on a