Home Assignment 1A

EITF41, Advanced Web Security

November, 2019

Solutions to eight A-assignments.

A-1

In EMV, SDA cards are cards that only support SDA data authentication. List some advantages and drawbacks of these cards compared to cards that support DDA. Can you find an attack that would work on SDA cards but not on DDA compatible cards?

SDA (static data authentication) is the simplest method for data authentication for the EMV standard, it checks data embedded in the card at the time of issuance and is therefore vulnerable to replay attacks since the authentication data (signature) sent to the terminal is never changed. That is, it is possible for a man-in-the-middle attack where another third party listens to the transfer of the data and uses this information to fraudulently authorize a new transaction through a copy of the chip data.

DDA (dynamic data authentication) is far more secure since it has an asymmetric key pair, instead it checks the data generated during the lifetime of the card. Every time authentication is made the terminal generates a random number which is included with the signed data. Since the signature will be different every time after authentication, replay attacks will prove futile. That is, the signature will only be valid for one authentication.

SDA was deployed before DDA and as such there are still cards using SDA. What is delaying the transition to DDA is mostly due to that it is more complex and demanding. For example it can take up to eight times longer to generate the cryptography on a DDA card. SDA does not require cryptographic processing by the chip, since it uses a "static cryptogram" permanently stored on the chip and as such is faster than the DDA counterpart.

A-3

Give two common ways to prove/make probable that the person making a card-not-present transaction is in physical possession of the card. Compare the two alternatives in terms of security.

Two common ways are either through AVS (address verification system) or by providing information existing on the card such as expiry date or the card security code CVV2/CVC2/CID (this also checks that the card is valid).

The billing address of the card is supplied by the customer and is not written on the card. As such supposedly only the customer is aware of the correct address. The merchant can also chose to only ship items to the billing address. In countries like Sweden where it usually is very easy to look up another private persons home address this might not be sufficient, but it does provide some sort of authentication of the cardholder.

Since the card security code is never used during CP (card present) transactions it is assumed that only the cardholder has access to the code. The code is a MAC based on the account number, version number and expiry date. This is somewhat more secure since the customer will have to have the physical card present (if they haven't memorized the security code). Though if the card is stolen or lost there is still a security risk since all needed information is provided on the card.

On their own they have some vulnerabilities, but integrated together they will be able to deny a lot more of fraudulent behaviours.

A-4

What is the difference between a three-party scheme and a four-party scheme for credit card payments?
In the three-party scheme (closed scheme) the parties involved are:

- Cardholder; The party making the online purchase.
- Merchant; The party selling items through a web store.
- Acquirer/Issuer.

In this model, the issuer (having the relationship with the cardholder) and the acquirer (having the relationship with the Merchant) is the same entity. This means that there is no need for any charges between the issuer and the acquirer. For example Diners Club International.

In the four-party scheme (open scheme) the parties involved are:

- Cardholder; The party making the online purchase.
- Merchant; The party selling items through a web store.
- Acquirer.
- Issuer.

In a four-party scheme, the issuer and acquirer are different entities, and this scheme is open for other members of the scheme to join in the competition. This signifies card schemes such as VISA and Mastercard. There is no limitations as to who may join the scheme, as long as the requirements of the scheme are met.

A-5

How does the Merchant verify the dual signature in SET?
The dual signature is a concept introduced with SET, which aims at connecting two information pieces meant for two different receivers :

- OI (order information) for merchant.
- PI (payment information) for bank.

The Cardholder sends its own certificate, dual signature, encrypted PI, PIMD and OI to the merchant in a purchase request message. The operation can be summarized as:

$$\text{DS} = \text{E}(\text{PR}_c, [H(H(PI) \mid\mid H(OI)])$$

where PRc is the customer's private signature key. Now suppose that the merchant is in possession of the dual signature (DS), the OI, and the message digest for the PI (PIMD). The merchant also has the public key of the customer, taken from the customer's certificate. Then the merchant can compute the quantities:

$$\text{H}(\text{PIMS} \mid\mid H[OI]); D(PUc, DS)$$

where PUc is the customer's public signature key. If these two quantities are equal, then the merchant has verified the signature.

The merchant verifies the dual signature using PIMD and OI. The merchant hashes the OI and sends an authorization request to the Payment Gateway. This message includes the encrypted PI, dual signature, OIMD, Transaction ID and the Cardholder's and Merchant's certificates. This is signed by the Merchant and symmetrically encrypted. The encryption key is encrypted with the Payment Gateway's public key.

A-8

How does the SET protocol provide non-repudiation?
Authentication is achieved through the use of digital signatures. Using a hashing algorithm, SET can sign a transaction using the sender's private key. This produces a small message digest, which is a series of values that "sign" a message. By comparing the transaction message and the message digest, along with the sender's public key, the authenticity of the transaction can be verified.

Digital signatures are aimed at achieving the same level of trust as a written signature has in real life. This helps achieve non-repudiation, as the consumer cannot later establish that the message wasn't sent using his private key.

Message digests can, using sufficient hardware to undertake a brute-force search of matching signatures, be forged, just like a real signature can be forged. This is unlikely, however, and does offer greater security because copying a signature and applying it to a different message will does not make it authentic.

A-10

What two general ways are there to enroll in VbV (3D Secure) and which would you say is better?
Enrollment can be done either as a separate registration process by logging in to the issuer and activating the card or in a face-to-face meeting with the issuer or in some other way that allows the issuer to authenticate the cardholder. Another way of enrollment is a method called Activation During Shopping, ADS. Then the enrollment is done as part of the first purchase made using VbV (Verified by Visa).

The main reason to use ADS is to simplify the enrollment process and to increase the adoption rate. Thus, this is solely a business choice made to attract more people to the product. As a consequence, security is compromised. First, in the middle of a purchase people are less likely to choose good passwords since their focus is on the purchase. Second, as pointed out above, sending sensitive information to a webpage that is difficult to authenticate is something people have been told over and over again not to do (since VbV will generate a pop-up when communicating with the issuer. For this reason I would say that enrollment during a separate registration process is better.

A-24

Explain how a hash chain, similar to the one computed in PayWord, can be used to implement a one-time-password login in e.g., Linux or Windows.
Say that the user has given the verifier the 1000th hash in a hash chain. When the user now wants to login the next time, the user will send the 999th hash, and the verifier will check that it is correct by hashing the 999th hash and see that it matches with the previously saved 1000th hash. After the verification is done, the verifier will instead store the 999th hash and expect the user to send the 998th, and so on.

If you want it even closer resemblance to how Payword uses hash chain, you can give the verifier the end of your chain, and every time you login you send the hash i steps away from the end, together with i. The verifier will then hash that hash i times and make sure it matches with the saved end of chain. Then next time, the user send the hash another step away, and so on.

A-32

Briefly explain how double-spending is prevented in Bitcoin. How can an attacker with huge computational resources perform double-spending?
To prevent double spending, without involving a bank, a transaction is put into a block, which in turn is part of a block chain. The block chain is known to all participants and all blocks in the block chain represents transactions that are valid and accepted by the network nodes. In order to add a new block to the block chain, a concept known as "proof of work" is used.

Several computers may compute a valid hash around the same time and broadcast it to the network. This means that the block chain will temporarily fork into two chains. This fork may continue for a while and the problem is solved by defining the chain that represent the most amount of work to be the valid chain. Thus, it is possible for a transaction to be included in a block that is later abandoned in favor of another block.

An attacker can attempt to change the history of his Bitcoin transactions by e.g., modifying a transaction such that less money is sent or money is sent to another address (and can therefore use it "again". This will require the attacker to produce a new valid block with the modified transaction and then create valid hashes for all subsequent blocks in the block chain such that the chain fork produced will be longer than that one used by the honest nodes in the network. For this to happen, the attacker must have computational power that exceeds the total computational power of the honest nodes. Otherwise, the modified chain will never be able to catch up with the real chain.