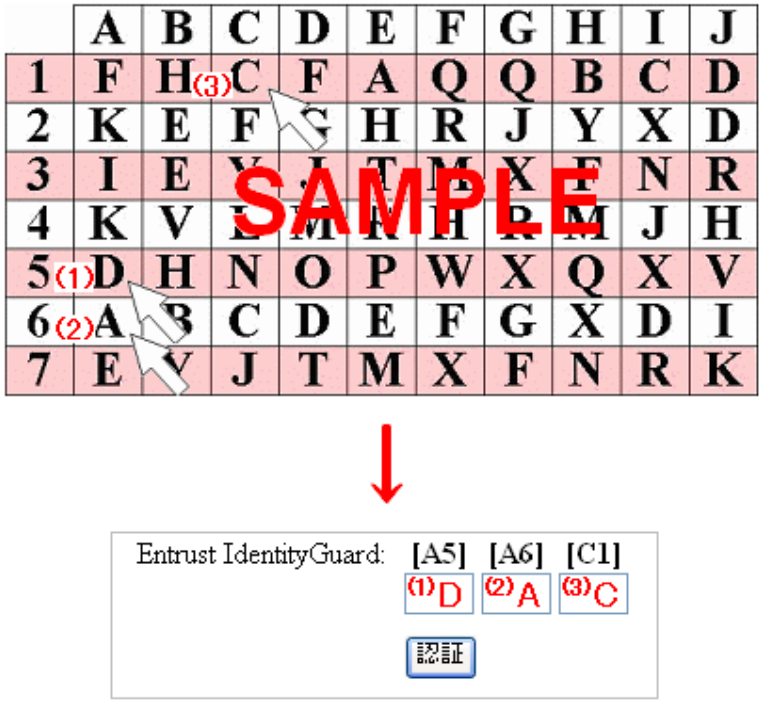
User Authentication threat landscape

User name + Password

- User authenticates with Username and Password
- Vulnerable to:
  - Password theft:
    - Phishing: the attacker steals user name and password using a fake website
    - Social Engineering: the attacker cheats the user to get user name and passwords (i.e. fake help desk call)
    - Shoulder Surfing: the attacker observes the user during authentication on the web portal and he steals credentials
  - Malware theft:
    - Keylogger: malware that monitors

Battle Card password

- User authenticates with Username and Password + a 2nd factor
- 2nd factor is a battle card matrix
- Vulnerable to:
  - Password theft:
    - Phishing: attacker asks the user to insert the (or part of the) matrix on a fake website
    - Social Engineering: attacker cheats the user and steals all the information in the matrix
    - Matrix theft
  - Malware theft:
    - malware collects matrix data every time the user make a transaction
    - malware redirects the user on a fake website that ask to insert the (or part of the) matrix
  - Man In The Browser: malware interacts with user's browser to modify online transactions and uses matrix data



One Time Password – SMS

- User authenticates with Username and Password + a 2nd factor
- 2nd factor is a one time password sent via SMS
- Vulnerable to:
  - Mobile phone theft: if the attacker steals the mobile phone and is able to get the password he will be able to act as the user
  - Mobile phone cloning
  - Malware theft:
    - Malware can steal and obfuscate SMS token
    - Emerging malwares are able to infect both PCs and smartphones at the same time (i.e. Eurograbber)
  - Man In The Browser (on PCs/Laptops): malware interacts with user's browser to modify online transactions and uses generated OTP

One Time Password – Email

- User authenticates with Username and Password + a 2nd factor
- 2nd factor is a one time password sent via email
- Vulnerable to:
  - Email account theft: if the attacker steals the email account and is able to get the password he will be able to act as the user
  - Malware theft:
    - Malware can steal and obfuscate emails containing the token (on PCs and smartphones)
    - Attacks based on Eurograbber attacking path can be implemented using email
  - Man In The Browser (on PCs/Laptops): malware interacts with user's browser to modify online transactions and uses generated OTP

One Time Password – Hardware Token

- User authenticates with Username and Password + a 2nd factor
- 2nd factor is a one time password generated by a hardware OTP token
- Vulnerable to:
  - Token theft: if the attacker steals the token and is able to get the password he will be able to act as the user
  - Man In The Browser: malware interacts with user's browser to modify transactions and steal OTP.

One Time Password – Software Token

- User authenticates with Username and Password + a 2nd factor
- 2nd factor is a one time password generated by software OTP token
- Even if the bank transaction is executed on the mobile smartphone where the soft token is installed current malware technologies are not able to steal the OTP
- Vulnerable to:
  - Token theft: if the attacker steals the token and is able to get the password he will be able to act as the user
  - Man In The Browser (on PCs/Laptops): malware interacts with user's browser to modify online transactions and uses generated OTP

How about another way ?

- Bob wants to send €100 to Alice
- Bob enters Alice's account number and amount of transfer
- Banking site receives transfer details
  - Calculates a challenge based on the transaction details
  - Sends challenge to Bob's browser ("383" for example)
- Bob uses a "calculator" provided by the bank
  - The transaction details are calculated on Bob's side and a "Response" is generated.
  - If the response is the same as the challenge then the transaction is authentic and accepted.

Transaction signing

- User provides a hash based on transaction summary (i.e. IBAN, transaction amount). This hash is verified on server side.
- This method guarantees that the transaction sent to the service (e.g banking portal) is the same that has been requested and verified by the user
- This attack is not vulnerable to MITB if the challenge is not calculated on the same device where the transaction is executed
- Public standard Implementation: OCRA (OATH Challenge Response Algorithm) - http://tools.ietf.org/html/rfc6287

How does it stop MITB attacks ?

- Bob wants to send €100 to Alice
- Bob enters Alice's account number and amount of transfer
- Attacker changes the transaction sending €1000 to Joe
- Banking site receives transfer details
  - Calculates a challenge based on **Joe's** transaction details
  - Sends challenge to Bob's browser ("**385**" for example)
- Bob uses a "calculator" provided by the bank
  - The transaction details are calculated on Bob's side and a "Response" is generated.
  - Response is "**383**", not same as "**385**" so transaction is blocked

Summary matrix

	Password Theft	Token/Device Theft	Malware Theft on PCs	Malware Theft on Smartphone/Tablet	Man in the browser
User name + Password	Vulnerable	N/A	Vulnerable	Vulnerable	Vulnerable
Battle Card	Vulnerable	Vulnerable	Vulnerable	Safe	Vulnerable
OTP SMS	Safe	Vulnerable	Vulnerable	Vulnerable	Vulnerable
OTP Email	Safe	Vulnerable	Vulnerable	Vulnerable	Vulnerable
OTP Hardware	Safe	Vulnerable	Vulnerable	Safe	Vulnerable
OTP Software	Safe	Vulnerable	Vulnerable	Safe	Vulnerable
Transaction Signing	Safe	Vulnerable	Safe (if the challenge is not calculated on the device where the transaction is executed)	Safe	Safe

Symantec VIP Strong Authentication

Symantec Soft Token Security

- Soft token protection:
  - Built-in OS data protection
  - Proprietary encryption and obfuscation to prevent copying of a credential from one device to another
  - Encryption Algorithm: AES with up to 1024 bits keys size
  - Additional security layers depending on device operating system

- OTP calculation algorithm:
  - Based on OATH standard

Symantec Transaction Signing

- Transaction signing algorithm:
  - Based on open standard OCRA (OATH Challenge Response Authentication)
- OCRA challenge can be calculated on hardware tokens or ad hoc mobile applications
- OCRA challenge is verified on Symantec datacenters through secure APIs
- Described by RFC6287

Thank you!

Christos Ventouris

cventouris@symantec.com