IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2015

BEng Honours Degree in Computing Part III
MEng Honours Degree in Electronic and Information Engineering Part IV
BEng Honours Degree in Mathematics and Computer Science Part III
MEng Honours Degree in Mathematics and Computer Science Part III
MEng Honours Degrees in Computing Part III
MSc in Advanced Computing
MSc in Computing Science (Specialist)
MRes in High Performance Embedded and Distributed Systems
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the Associateship of the City and Guilds of London Institute

PAPER C331

NETWORK AND WEB SECURITY

Tuesday 24 March 2015, 14:00
Duration: 120 minutes

Answer THREE questions

Paper contains 4 questions
Calculators not required

1 **Public Key Cryptography**

Let p be a large prime number, g a generator of the multiplicative group $\mathbb{Z}_p^* = \{1, 2, \ldots, p-1\}$, and $H: \mathbb{Z}_p^* \to \{0,1\}^n$ a hash function. Private keys x are random elements of $\mathbb{Z}_p^*$, and the public key y corresponding to such an x equals $y = g^x \bmod p$.

Messages m have bitlength n and are encrypted with a random k from $\mathbb{Z}_p^*$ to ciphertext $c = (c_1, c_2)$ defined by

$$c_1 = g^k \qquad c_2 = m \oplus H(y^k) \tag{1}$$

whereas decryption of a pair (c_1, c_2) is given by

$$c_2 \oplus H(c_1^x) \tag{2}$$

All exponentiations in (1) and (2) are modulo p.

a i) State what a decryptor needs to know to apply decryption in (2).

ii) Show that decryption of (c_1, c_2) recovers the plaintext m.

iii) Briefly discuss the roles of the c_1 and c_2 part in this cryptosystem.

b Sketch a protocol in which Alice and Bob exchange a session key of n bits by using the above public cryptosystem, detailing what is private to which parties.

c Let $h: \{0,1\}^* \to \{0,1\}^n$ be a hash function and let $bin(n)$ denote the binary encoding of elements n in $\mathbb{Z}_p^*$.

i) Use h and bin to construct the above hash function $H: \mathbb{Z}_p^* \to \{0,1\}^n$.

ii) Discuss which security properties of h – preimage resistance, collision resistance, and second preimage resistance – your hash function H from item i) would or would not inherit from h.

The three parts carry, respectively, 35%, 30%, and 35% of the marks.

2a **Attack Model for symmetric block cipher.**

1. A secret key K is generated and given to the oracle, and the attacker Eve has no knowledge of K.

2. Eve chooses any plaintext m she wishes and sends it to the oracle. The oracle answers with ciphertext of m encrypted under key K but won't perform any decryptions. Eve can execute step 2 only once.

3. Eve generates messages m_0 and m_1 of the same length and gives them to the oracle. The oracle chooses b in $\{0,1\}$ and returns to Eve the ciphertext of m_b encrypted under key K.

4. Eve now has to guess the value of b.

Eve wins this attack game if there is some $\epsilon > 0$ such that the probability of Eve guessing b correctly is $\geq 0.5 + \epsilon$.

i) Does Eve win this game if the block cipher is AES used the ECB mode? Justify your answer: for which ϵ can she win or why is there no such ϵ.

ii) Let AES be used in CBC mode. Assume that the IV is initially the block of zeros and simply incremented each time a new message is encrypted in CBC mode. Can Eve win this game? Justify your answer as in part i).

b **Key Management and Access Control.** In a system involving a third trusted party TTP, files are only stored in encrypted form, based on functions

$$owner\colon Files \to Users \qquad control\colon Files \to KeyControls$$
$$share\colon Files \to KeyShares$$

where $owner(f)$ is the user who owns file f, $share(f)$ is a keyshare for file f only known to $owner(f)$ and TTP, and $control(f)$ is the key control only known to TTP. A file f is encrypted and decrypted with key $share(f) \oplus control(f)$.

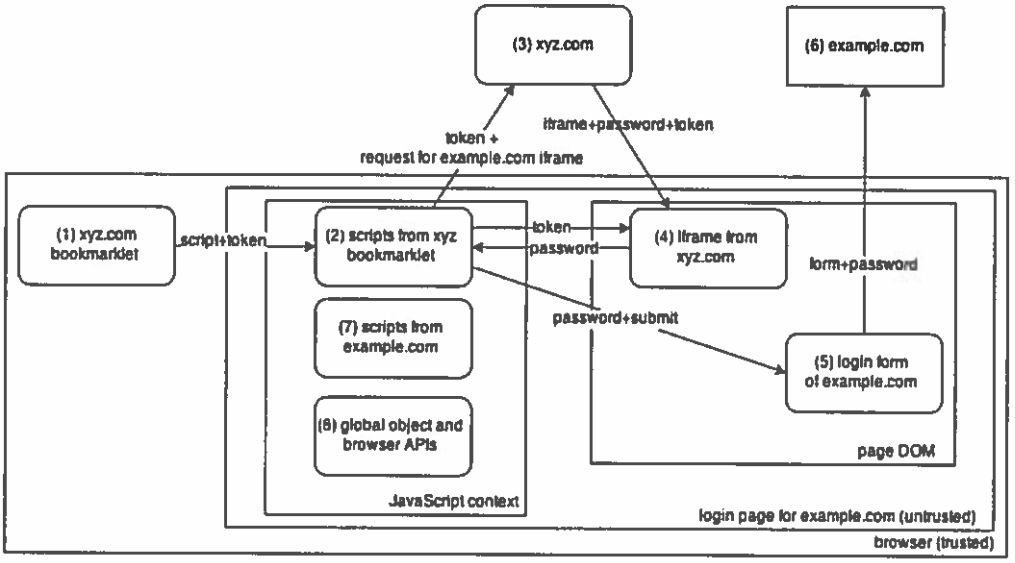
i) Describe a protocol in which the owner of a file requests information needed for constructing the key of that file.

ii) Design key management in this system such that $owner(f)$ may invite other users to gain the capability of decrypting file f.

iii) Extend your design to allow for the revocation of encryption keys based on either time stamps or owner requests.

The two parts carry equal marks.

3 Company XYZ provides a password manager that is accessible as a bookmarklet. Users sign up for the service at `xyz.com`, add credentials for selected websites to their account, and are given one bookmarklet for signing in to the selected websites. The bookmarklet contains a secret, unguessable token that uniquely identifies the user on `xyz.com`. Consider the diagram below, whose elements are referenced in brackets. When a user wants to sign in to a website, she visits its login page and clicks on the bookmarklet (1). The bookmarklet script is executed on the login page (2), and loads from `xyz.com` (3) an iframe (4) which contains the user's token and her password for the visited website. The bookmarklet script sends its token (using `postMessage`) to the iframe. The iframe sends the password to the bookmarklet script. The bookmarklet script injects the password in the login from (5), and submits the form (6). The main security goals are that the password for any site A should only be submitted to the site A itself, and only as a result of the user clicking on the bookmarklet while visiting A. In particular, the proposed design should protect from network attackers and prevent malicious login pages from stealing passwords for other websites.



a You are asked to perform a threat analysis for the XYZ password manager.

i) Briefly describe each threat category denoted by the STRIDE acronym.

ii) Describe four security threats for the password manager application. Ignore the threats to account management and bookmarklet installation. Assume that the web browser is trusted and cannot be compromised.

iii) Propose a mitigation for each threat identified in the answer to part ii) or, if mitigation is unfeasible, discuss another way to address the threat.

b You are asked to do a code review for the password manager bookmarklet. Consider the code fragment below, which is part of (2) in the diagram.

```
1 var CryptoAPI = (function(){
2  var encoding = {...}; % assume secure
3  var API = {
4   sha1: {
5    name: 'sha1',
6    identifier: '2b0e03021a',
7    size: 20,
8    block: 64,
9    hash: function(s)
10   {
11    var len = (s+='\x80').length, blocks = len >> 6,
12        chunk = len&63, res = "", i = 0, j = 0,
13        H = [0x67452301, 0xEFCDAB89, 0x98BADCFE, 0x10325476, 0xC3D2E1F0],
14        w = [];
15    while(chunk++ != 56)
16    {
17     s+="\x00";
18     if(chunk == 64){ blocks++; chunk = 0; }
19    }
20    for(s+="\x00\x00\x00\x00", chunk=3, len=8*(len-1); chunk >= 0; chunk--)
21     s += encoding.b2a(len >> (8*chunk) &255);
22    for(i=0; i < s.length; i++)
23    {
24     j = (j<<8) + encoding.a2b(s[i]);
25     if((i&3)==3){ w[(i>>2)&15] = j; j = 0; }
26     if((i&63)==63) CryptoAPI.sha1._round(H, w);
27    }
28    for(i=0; i < H.length; i++)
29     for(j=3; j >= 0; j--)
30      res += encoding.b2a(H[i] >> (8*j) & 255);
31    return res;
32   }, % end hash
33   _round: function(H, w){...} % assume secure
34  }, % end sha1
35  ... % definition of other fields of API object: assume secure
36 return API; % end of body of anonymous function
37 })() % end CryptoAPI
```

i) Identify three vulnerabilities that lead to execution of arbitrary JavaScript code.

ii) Propose a fix for each vulnerability identified in part i). (For example, you may answer: "In line *n*, replace [vulnerable code] with [fixed code]".)

iii) Write a short proof-of-concept exploit for each vulnerability identified in part i). (For example, you may write some lines of code to execute before or after loading the `CryptoAPI`, or show a way to call a `CryptoAPI` function which will cause undesired behaviour.)

The two parts carry equal marks.

4 Cross-site request forgery (CSRF) consistently ranks among the CWE/SANS Top 25 Most Dangerous Software Errors and in the OWASP Top 10 list of web application security flaws. CSRF is a particularly critical vulnerability because it can subvert sessions, which are key components of web applications.

a Let us focus on attackers for now.

i) List the main low-level capabilities available to a web attacker. Briefly describe two scenarios where a user browsing the web is exposed to such an attacker. (Note: you are not asked to describe details of the attacks.)

ii) As for item i), but for an active network attacker.

iii) As for item i), but for a related-domain attacker.

iv) What is the main difference between a related-domain attacker controlling a subdomain of a target website, and an attacker that is able to launch a stored XSS attack on the same subdomain of the target website?

b Consider the following scenario S. A user visits `login.example.com`, where she holds an account. The user signs in, and the server redirects her browser to `example.com`, where she is presented with a page containing her personal data and a form. The user inserts some data in the form and submits the form. The server receives the data and, if the user is still signed in, the server stores the data on behalf of the user.

i) With reference to scenario S above, describe in a few bulletpoints the simplest session mechanism you can think of that defends the website against CSRF attacks launched by a web attacker. Briefly discuss the security of your solution, considering how this kind of attacker would try to break session integrity.

ii) As for item i), but for the the active network attacker.

iii) As for item i), but for a related-domain attacker on `attacker.example.com`.

The two parts carry, respectively, 40% and 60% of the marks.