Final exam in

Web Security EITF05

Department of Electrical and Information Technology
Lund University

October 24th, 2012, 8.00-13.00

- You may answer in either Swedish or English.
- If any data is lacking, make (and state) reasonable assumptions.
- Use legible hand writing. If your answers cannot be read, you will receive zero points on that problem.
- Grading is done as follows.
  Grade 3 = 20–29 points,
  Grade 4 = 30–39 points,
  Grade 5 = 40–50 points.

Good luck!

Martin, Paul & Christopher

Problem 1. Why does DMARC protect well against phishing attacks, but not against spam in general? (3 points)

Problem 2. Assume that a web page is using the php.ini directives

```
session.use_only_cookies=0
session.use_trans_sid=1
```

Moreover, the session ID is not regenerated when users log in. Describe an attack that has a fair chance of succeeding in this scenario.
HINT: The `session.use_trans_sid=1` tells the server to rewrite all URLs so that they additionally include the session ID. (3 points)

Problem 3. Explain what file inclusion is and how it can be exploited by an adversary to run his or her customized PHP code on a vulnerable server. (3 points)

Problem 4. Describe how Cross-Origin Resource Sharing (CORS) allows an application to violate the same-origin policy. You do not have to describe pre-flight requests. Just explain how the `origin` and `Access-Control-Allow-Origin` headers are used in a simple GET request. (3 points)

Problem 5. A yellow pages company wants to make sure that their spider crawls all subdomains of the entire .se domain. Assume that the DNS server of the .se domain implements DNSSEC. The yellow pages company can then use NSEC records for zone walking so that all existing subdomain names can be retrieved from the DNS. Explain how zone walking works. (3 points)

Problem 6. Give the Base64 encoding of the word "SPAM".

Hint: Decimal representation of ASCII characters is given by:

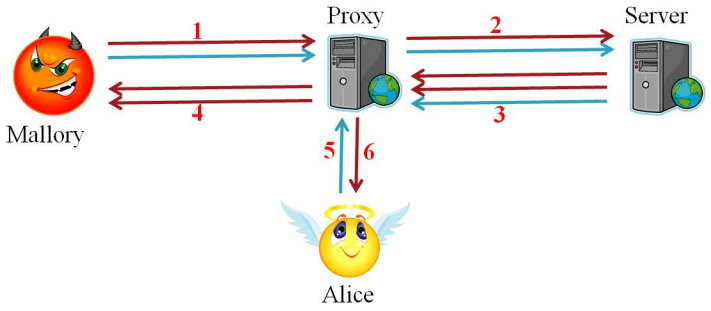
$$A = 65, B = 66, \ldots, Z = 90, a = 97, b = 98, \ldots, z = 122$$

The Base64 alphabet is:

$$0 = A, \ldots, 25 = Z, 26 = a, \ldots, 51 = z, 52 = 0, 53 = 1, \ldots, 61 = 9, 62 = +, 63 = /$$

(3 points)

Problem 7. How can an adversary realize a phishing attack using HTTP response splitting? Specifically explain what is needed for the attack to be successful. You may refer to the picture below.



(3 points)

Problem 8. Briefly describe (a sentence or two) each of the following document types/languages/techniques and in which context they are used; XML, HTML, CSS, JavaScript, PHP and AJAX. (3 points)

Problem 9. The yellow pages company from Problem 5 wants to collect phone numbers by using a spider to crawl all web content of the entire .se domain, and then match the content with a suitable regular expression. You are hired to suggest a regular expression that matches swedish phone numbers; a 2–4 digit area code beginning with a 0, followed by an optional dash, followed by 5–7 digits that may be arbitrarily space-separated. The following examples should be matched.
HINT: The space character can be represented by a literal space.

01-234 56 78	012-34 56 78	0123-456 78
01-2345678	012-345678	0123-45678
012345678		

(3 points)

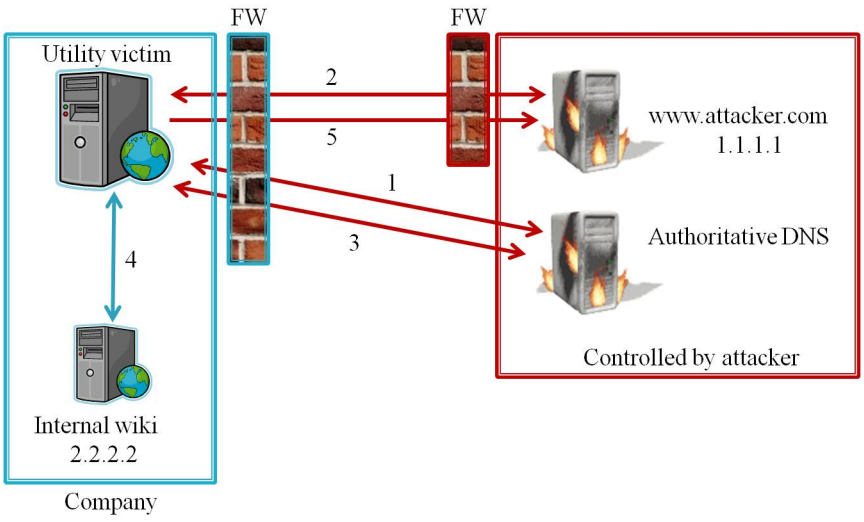
Problem 10. Cross Site Scripting (XSS) is one of the most common vulnerabilities on the web.

a) What is the difference between persistent and non-persistent XSS attacks?

b) How can XSS be used to steal a session cookie? Why does this not violate the same-origin policy?

(1+2 points)

Problem 11. Explain how an adversary can use DNS rebinding to breach a company firewall and access intranet web pages. Specifically explain what is needed for the attack to be successful, and how the same-origin policy is violated. You may refer to the picture below.



(5 points)

Problem 12. The first version of Digest Access Authentication (RFC 2069) created the response as

$$\mathrm{MD5}(\mathrm{MD5}(A_1) : nonce : \mathrm{MD5}(A_2)) \tag{1}$$

where

$$\begin{aligned} A_1 &= username : realm : password \\ A_2 &= method : URI \end{aligned}$$

Now, assume that an attacker, in a man-in-the-middle scenario, can choose the nonce before forwarding the HTTP response to the user-agent. Upon intercepting the request with the response field computed as in (1), the attacker performs a time-memory-tradeoff attack to recover the password. The tables used were downloaded from the Internet and covered 2^6 usernames, 2^6 realms, 2^{50} passwords and required disk space corresponding to $M = 2^{37}$ in the tradeoff curve $N^2 = M^2T$, where N is the size of the search space and T the online time required in the attack. The method was fixed to GET and the URI was fixed to `/index.html`.

a) Describe how the table chains were constructed. You can describe it using either Hellman tables or Rainbow tables. You do not explicitly have to provide the reduction function.

b) How much time was spent in the offline phase of the attack, i.e., constructing the tables?

c) How much time was spent in the online phase of the attack?

d) How much time did the attacker save by performing the TMTO attack compared to a brute force attack on the password.

(2+1+1+1 points)

Problem 13. Consider the following SMTP header:

```
Received: from tonallan.com (178-223-104-216.dynamic.isp.telekom.rs. [178.223.104.216])
        by mx.google.com with SMTP id go13si33901881bkc.9.2012.10.08.10.50.52;
        Mon, 08 Oct 2012 10:51:07 -0700 (PDT)
```

a) Explain the different parts of the header.

The header string added when using hashcash is given by

```
ver:bits:date:resource:[ext]:rand:counter
```

b) Explain the algorithm used when constructing the header. In particular, focus on the `bits` and `counter` fields.

c) How difficult is it to construct a valid header?

(2+2+1 points)

Problem 14. Briefly explain the following terms.

a) Prepared statement

b) Birthday paradox

c) DNS amplification

d) CSRF

e) Statistical filtering

(5 points)
