Final exam in

Web Security EITF05

Department of Electrical and Information Technology
Lund University

November 2nd, 2018, 14.00–19.00

- 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!

Paul

Problem 1. Consider an SQL injection attack.

a) Write some PHP code and use it to illustrate and explain how an SQL injection attack works.
b) Motivate why this is potentially the most dangerous attack for any company.
c) How does the same origin policy protect against SQL injection attacks? (3 points)

Answers

a) Be creative. Code syntax is not the focus, but rather your principal explanation.
b) Sensitive data from each and every customer may leak, causing irreparable damage to the company brand.
c) It does not.

Problem 2. Give a regular expression that matches an IP address (IPv4). The following variations should match;

```
127.0.0.1
255.255.255.255
0.0.0.0
```

but not

```
256.256.256.256
123.456.789.012
```

Matching leading zeros is optional. (3 points)

Answer
One possibility:

```
^(?:(?:25[0-5]|2[0-4][0-9]|[01]?[0-9]?[0-9])\.){3}
    (?:25[0-5]|2[0-4][0-9]|[01]?[0-9]?[0-9])$
```

Problem 3. Consider a DNS server that implements Domain Name System Security Extensions (DNSSEC).

a) How many signatures does the DNS server need to generate on-the-fly for each DNS request it receives? Motivate.

b) NSEC allows zone walking. What is zone walking, and how (explain the idea briefly) is this prevented in NSEC3?

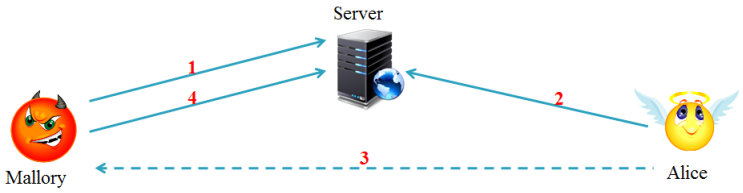
Answers

a) None. Signatures are pre-generated for efficiency. A DNSSEC-enabled DNS uses the interval trick to achieve this (sort all available domain names lexicographically and pre-sign all successive pairs of entries).

b) Zone walking is when an external party creates a list of all (sub)domains served by a given DNS server. NSEC3 uses the interval trick, but replaces lexicographical sorting by sorting the hashed names instead.

(1.5+1.5 points)

Problem 4. Consider the following illustration of an XSS attack with three involved entities; Mallory, Server and Alice.



a) Does TLS protect against XSS attacks? Motivate.

b) What provides good protection against XSS attacks? Motivate.

Answers

a) No. XSS attacks are application layer attacks, so they work even if TLS is applied.

b) Applying Content Security Policy (CSP) rules is a good countermeasure. It is also a good idea to apply filtering to all data that can involve input from an external entity (user).

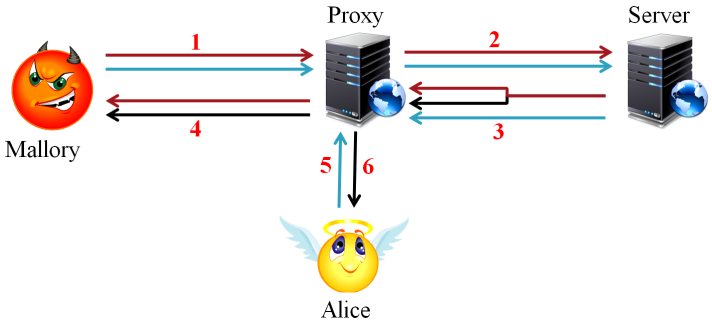
(1.5+1.5 points)

Problem 5. Explain how Domain-based Message Authentication, Reporting and Conformance (DMARC) works.

Answer

See lecture notes. (3 points)

Problem 6. Consider the following illustration of an HTTP response splitting attack.



a) Briefly explain how an HTTP response splitting attack works. You may refer to the picture.

b) In the attack, what is the purpose of the Proxy?

Answers

a) See lecture notes.

b) The Proxy needs to be a caching server for this attack to be meaningful. Otherwise the request is simply served to Mallory, which is pointless since she already knows what her own malicious page looks like. However, if the malicious page is cached, it can be served to other users (Alice) as well.

(2+1 points)

Problem 7. A DKIM signature header of an email is given below.

```
DKIM-Signature:
v=1;
a=rsa-sha256;
c=simple/relaxed;
d=gmail.com;
s=gamma;
h=received:message-id:date:from:to:subject:mime-version:content-type;
bh=9gicsZnlcLK7yYh6VIrgyAMMRZiWsSbWqSPIhc78RRk=;
b=k4ofvpHPkaQmvuSoGVhRrnCsPK+JEuv9KUrZO7aiypvf/6Y1N2iIatvLvdzwOnZX
  /W6Kxyx6Z4Ybuk8Dqk/vNTIE7Jpy+GQUUHFvM0NFtmZo1CbGRvo8DdHnXRBB/qWw
  lV+Z6wxw/mq7lNuJknVprOAaTLws5mwcZ+AWL8KwHg0=
```

a) Does DKIM support usage of more than one public key per domain? Motivate.

b) How does DKIM provide integrity protection, and for which parts of the email? (What is signed, and how?)

Answers

a) The selector specifies which public key should be used for verification. One domain can use several different keys for signing, so the selector is used to distinguish the different keys.

b) `bh` is a hash of the body. The DKIM header (including `bh`) is included in `h`, even though it is not explicitly listed. Thus, the signature in `b` covers `bh` as well. All signed headers and the message part are covered.

(1.5+1.5 points)

Problem 8. An engineers has censored some famous quotes using Base64. What did they say? Choose any **one**. Show your calculation.

Margaret Atwood, The Blind Assassin:
The best way of `a2VlcGluZw==` a secret is to pretend there isn't one.

George Orwell, 1984:
If you want to keep a secret, you must also hide it from `eW91cnNlbGY=`.

Benjamin Franklin, Poor Richard's Almanack:
Three may keep a secret, if two of them are `ZGVhZA==`.

Hint: Decimal representation of ASCII characters:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122

The Base64 alphabet:

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51

0	1	2	3	4	5	6	7	8	9	+	/
52	53	54	55	56	57	58	59	60	61	62	63

(3 points)

Answers

a) keeping
b) yourself
c) dead

Problem 9. Bitcoin uses a hasing technique that is very similar in functionality to that used in Hashcash. Consider a Hashcash solution in which a string

$$ver : bits : date : resource : rand : counter$$

is hashed using SHA-1, where

ver is version number (currently 1),
bits indicates how costly the function is for sender,
date gives current date,
resource is recipients email address,
rand is a random number.

a) How is a proper *counter* value determined?
b) How many times must the hash function be invoked to *generate* a valid Hashcash header with $bits = 30$? Exactly or on average?
c) How many times must the hash function be invoked to *verify* a valid Hashcash header with $bits = 30$? Exactly or on average?

Answers

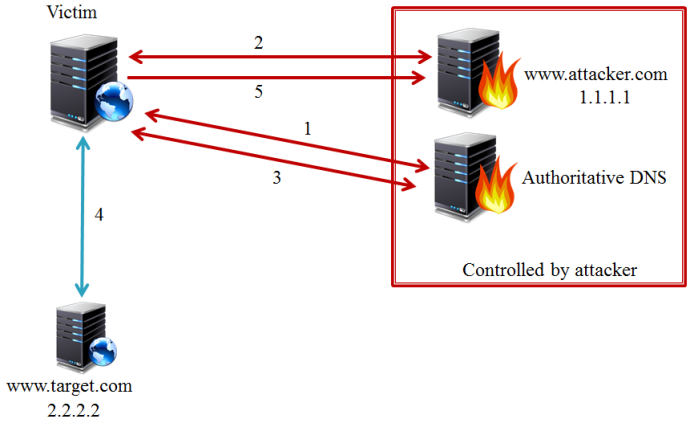
a) The HashCash header is valid if the hashed string has *bits* initial zeros. You can create a valid HashCash header as follows. Initially construct the string with the *counter* part set to zero. While that string does not hash to a value with *bits* initial zeros (while it is not a valid HashCash string), increase *counter*.
b) About 2^{30}.
c) Exactly once.

(3 points)

Problem 10. Consider the following illustration of a DNS rebinding attack.

a) Will the attack work if there is a firewall between the Victim and the Attacker? Motivate.

b) How would step 3 differ if the Victim's browser implements DNS-pinning?



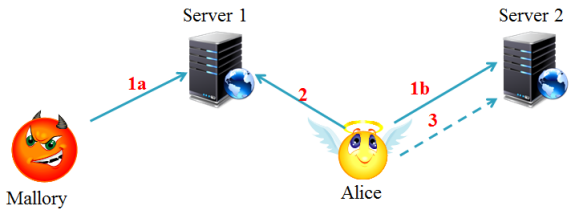
Answers

a) Yes. The Victim can initiate the attack from behind a company firewall, the browser will still send requests and receive responses.

b) The Victim's browser would ignore the short TTL provided by the Attacker's DNS server in step 1 and make the next request (arrow 4) to 1.1.1.1 instead. It would do this directly without re-resolving the domain name (skipping step 3).

(1.5+1.5 points)

Problem 11. Consider the following illustration of a CSRF attack.



a) Briefly explain how a CSRF attack works. You may refer to the picture.

b) Explain how CSRF protection with synchronizer token pattern works.

Answers

a) See lecture notes.

b) A CSRF token (fresh nonce) is embedded in a hidden field in the web page. This CSRF token is also stored server-side. When the user sends a request, the CSRF token is included. When the request reaches the server, the request is validated by comparing the CSRF token in the incoming request with the CSRF token value stored server-side. Mallory cannot access the page content, so she cannot construct a valid request (step 3).

(2+3 points)

Problem 12. HTTP digest authentication (RFC2617) is a challenge response protocol in which the client calculates the digest (the response) according to

$$\text{MD5}(\ \text{MD5}(A1) : nonce : nc : cnonce : qop : \text{MD5}(A2)\ ),$$

with

$$\begin{aligned} A1 &= username : realm : password, \\ A2 &= \begin{cases} method : URI & \text{if } qop = auth, \\ method : URI : \text{MD5}(entity\text{-}body) & \text{if } qop = auth\text{-}int. \end{cases} \end{aligned}$$

a) Explain the usage and purpose of the *realm* parameter?

b) Why does the *cnonce* parameter protect against TMTO attacks?

c) The *nonce* parameter does not protect against TMTO attacks, why?

Answers

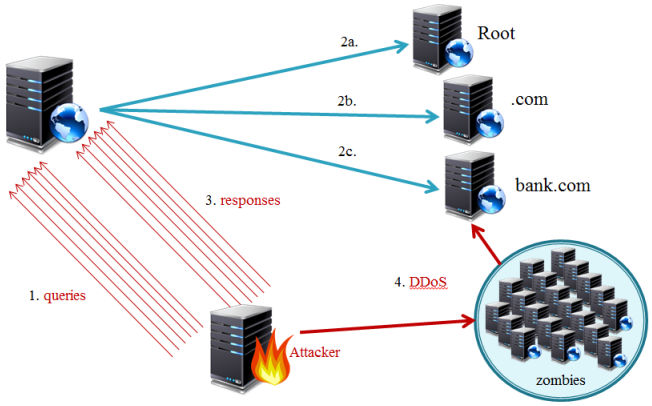
a) The *realm* parameter is a text string that explains to the user which password to use. One site could potentially use several different passwords to protect (different) resources.

b) The *cnonce* is chosen by the client after she has received the challenge, and *cnonce* is also included as a parameter when computing the response. A MITM cannot simply replace the *cnonce*, since it is never passed as part of the challenge, and she cannot re-calculate the response with a different *cnonce* of her own choosing.

c) The *nonce* is generated server-side and is included in the challenge sent to the client. The challenge is not stored server-side, but rather just echoed in the response. This makes it possible for a MITM to replace *nonce* with another value that she has chosen.

(1+2+2 points)

Problem 13. Consider the following illustration of a DNS cache poisoning attack. The success rate of the attack depends on how many queries and responses that can be sent in steps 1 and 3 (before the first response in step 2c has been delivered).



a) How is the birthday paradox leveraged in the DNS cache poisoning attack?

b) What is the purpose of the botnet?

c) How would usage of TCP (instead of UDP) provide protection? Motivate.

Answers

a) At least one of the transaction IDs in the fake responses sent in step 3 must be matched with at least one of the (randomly chosen) transaction IDs in step 2c.

b) The purpose of the botnet is to make responses from the `bank.com` DNS take longer. This gives the attacker more time for sending fake responses.

c) This would make DNS amplification attacks much more difficult to leverage. When using UDP, the attacker simply writes the desired destination IP address in the request. With TCP, this is no longer possible due to the handshake procedure.

(2+1+2 points)

Problem 14. Briefly explain the following terms and acronyms.

a) SMTP
b) Reflected XSS
c) CORS
d) Digest HTTP authentication
e) HEAD (the HTTP method)

Answers

a) Simple Mail Transfer Protocol, a protocol for transmitting email from a sender to a destination email address.
b) A non-persistent XSS attack. An XSS attack in which the payload is included in a link that the victim clicks.
c) Cross-Origin Resource Sharing, a protocol for handling web-based resource sharing between different domains. Useful for making mashups, displaying content from several different domains (weather, news, stock market stats,...).
d) An HTTP authentication protocol for protecting resources from general access. User must provide pre-approved username and password for access.
e) HTTP method that does the same thing as GET, except that it returns only the header part of the response.

(5 points)
