

Basic Web Attacks—Holiday Hack Trail—KEY

Easy Mode

Questions

- 1) How did you change the request so that you won the game?

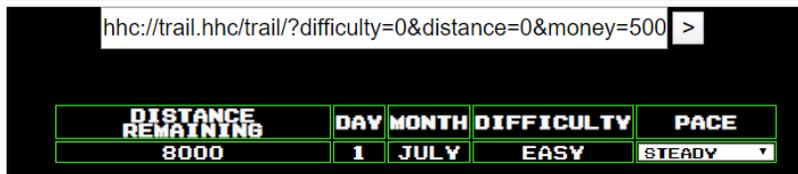
The simplest way is to click distance in the URL and change it to 8000. Note that the top left of the web page displays the distance left to go and the browser sends distance traveled to the site. Distance left to go plus distance traveled = 8000

- 2) What could the web designer do to prevent you from cheating this way?

They can make tricky algorithms in the browser to make changes easy to detect. When we get to Hard mode, the developer has implemented an algorithm to detect changes. The problem is that any algorithms in the browser are available to the attacker as well and can be broken. The best way is to keep the official copy of the data on the server.

- 3) What was the request method the browser used?

The challenge designer has crafted the site to look like it is using the GET method.



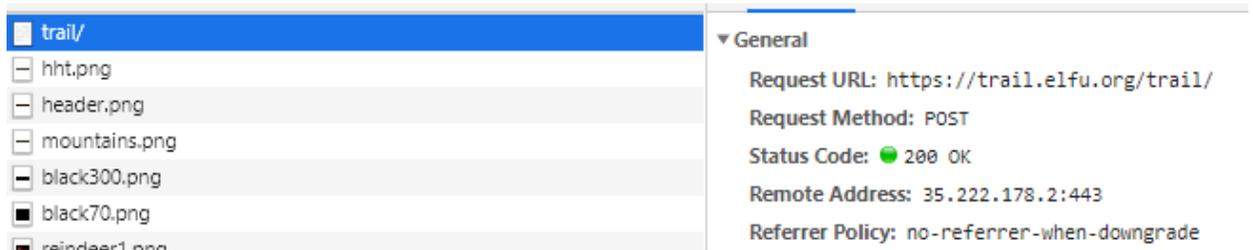
However, if you look behind the scenes, he is using POST. The correct answer for the students is GET.

Medium Difficulty Mode

Questions

- 4) What was the HTTP Request Method the browser used? (GET, HEAD, POST, CONNECT, and TRACE are possible answers.)

POST



5) What status code did the server return?

200 OK. It means everything is fine, here's your data.

6) What is the IP address of the web server?

35.222.178.2

Questions

7) What did you do to win the game?

Change distance to 8000 and click GO.

```
</button>
<button type="submit" class="btn" name="action" value="go" onmouseover="reindeerwaggle()" onmouseout="reindeerwaggle()">Go</button>
<br>
<br>
<div id="statusContainer">
  <input type="hidden" name="difficulty" class="difficulty" value="1">
  <input type="hidden" name="money" class="difficulty" value="3000">
  <input type="hidden" name="distance" class="distance" value="8000">
  <input type="hidden" name="curmonth" class="difficulty" value="8">
  <input type="hidden" name="curday" class="difficulty" value="1">
```

Distance before and after change—0 before and 8000 after.

HARD Mode

Question

8) What is, most likely, the hash function that created the hash in the image above?

google.com/search?q=commonly+used+hash+function+that+creates+an+output+that+is+128+bits+long.&

commonly used hash function that creates an output that is 128 bits long.

All Shopping Images Videos News More Settings Tools

About 1,730,000 results (0.82 seconds)

MD5

MD5 is the Message Digest **algorithm 5**. It is the **most widely used** of the MD family of **hash** algorithms. MD5 **creates a 128-bit hash** value based on any input length. MD5 has been quite popular over the years, but there are weaknesses where collisions can be found in a more practical amount of time.

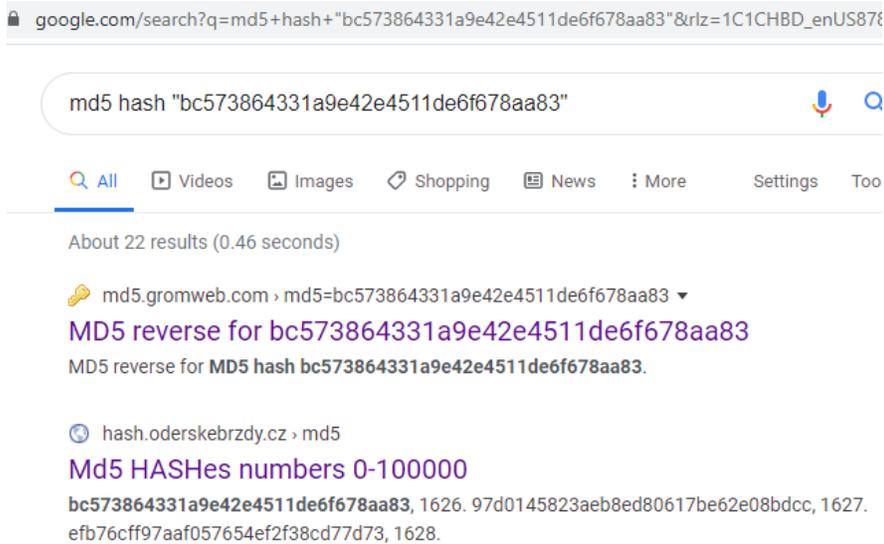
en.wikipedia.org

www.sciencedirect.com › topics › computer-science › one-way-hash-fun...
[One-Way Hash Function - an overview | ScienceDirect Topics](#)

MD5 is the hash algorithm

Question

- 9) What was the input to the hash function, so that "bc573864331a9e42e4511de6f678aa83" was the output?



1626 was the input to the hash.

Questions

- 10) What is the relationship between the change in distance and the change in number that is hashed?

After clicking GO several times at the beginning of the game, I noticed that Distance, Day, and Food were usually the only parameters that changed. That prompted me to make this spreadsheet.

Distance	Day	Food	Hash	Cracked	dist delta	day delta	food delta	total delta	cracked delta
0	1	100	bc573864331a9e42e4511de6f678aa83	1626	0	0	0	0	
34	2	92	b147a61c1d07c1c999560f62add6dbc7	1653	34	1	-8	27	27
82	3	84	26751be1181460baf78db8d5eb7aad39	1694	48	1	-8	41	41
127	4	76	b29eed44276144e4e8103a661f9a78b7	1731	45	1	-8	38	37
127	5	68	62889e73828c756c961c5a6d6c01a463	1724	0	1	-8	-7	-7
interesting, lost a runner on day 4, which made the total and cracked deltas differ by one									
each runner must add one to the hash.									

change in hashed number = change in distance (as long as nothing else changes)

- 11) How did you beat the game?

To make it more interesting, I clicked GO a couple of times to get a distance to go that was not 8000.

Original Distance = 62

```

<button type="submit" class="btn" name="action" value="go" onmouseover="
reindeerwaggle()" onmouseout="reindeerwaggle()">Go</button>
<br>
<br>
<div id="statusContainer">
  <input type="hidden" name="difficulty" class="difficulty" value="2">
  <input type="hidden" name="money" class="difficulty" value="1500">
  ... <input type="hidden" name="distance" class="distance" value="62"> == $0
  <input type="hidden" name="curmonth" class="difficulty" value="9">

```

The hash right now is 757f843a169cc678064d9530d12a1881

```

<input type="hidden" name="meds" class="meds" value="2">
<input type="hidden" name="food" class="food" value="84">
<input type="hidden" name="hash" class="hash" value="
"757f843a169cc678064d9530d12a1881">
</div>
</table> </table>

```

Change distance to 8000

```

<div id="statusContainer">
  <input type="hidden" name="difficulty" class="difficulty" value="2">
  <input type="hidden" name="money" class="difficulty" value="1500">
  .. <input type="hidden" name="distance" class="distance" value="8000"> == $0
  <input type="hidden" name="curmonth" class="difficulty" value="9">
  <input type="hidden" name="curday" class="difficulty" value="3">
  <input type="hidden" name="name0" class="name0" value="Lila">

```

The original hash was 757f843a169cc678064d9530d12a1881. Cracked, that's 1674.

```

Holiday Hack Trail x Md5 HASHes numbers 0-100000 x +
Not secure | hash.oderskebrzdy.cz/md5.php?kolik=0-100000
148510031349642de5ca0c544f31b2ef 1670
647c722bf90a49140184672e0d3723e3 1671
2451041557a22145b3701b0184109cab 1672
a0f3601dc682036423013a5d965db9aa 1673
757f843a169cc678064d9530d12a1881 1674
64f1f27bf1b4ec22924fd0acb550c235 1675
831c2f88a604a07ca94314b56a4921b8 1676

```

Distance changed by 8000 – 62 = 7938

Distance increased by 7938, so hashed number needs to increase by 7938

1674 + 7938 = 9612

Hash 9612 to get new hash

```

002d40e0a300d91000261312c2493004 9011
85b6c99bb36d6e7be78bf8fd28d6e43d 9612
035734a664004fb0216f1e45a4758f1f 0613

```

Put 85b6c99bb36d6e7be78bf8fd28d6e43d in for the new hash.

```



```

Click GO



- 12) Can you think of a way that this vulnerability could be fixed? Remember that any code on the browser can be studied or changed by the attacker.

Keep the official copy of the data on the server. Don't trust the user or the browser!!